

# 大学生数学建模 竞赛指南

主 编 肖华勇

副主编 周吕文 赵 松

電子工業出版社·

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书是一本指导大学生全方位备战数学建模竞赛的辅导书,从多角度介绍了数学建模及相关竞赛的背景知识;按照参赛流程解答了数学建模竞赛的常见问题;介绍了数学建模竞赛中常用的软件;讲解了数学建模的常用模型;精选了典型赛题进行详解;邀请了获奖学生和指导教师分享成功经验;介绍了数学建模竞赛过程中常用的网站。

本书在解答数学建模竞赛中的常见问题时,不仅解答了组建团队、赛前准备和时间安排等问题,还解答了文献检索、撰写论文及论文排版的相关问题,旨在使读者对数学建模的整个流程有非常清晰的认识。

本书不仅介绍了历年数学建模竞赛中常用的方法,分析了相关的赛题,还详解了实现的程序代码,让学生真正做到学以致用,而不是纸上谈兵。本书还邀请了获奖参赛队员和优秀指导教师,从不同的角度分享比赛中的成功经验,为参赛学生和教师提供不同角度的参考。

本书主要面向参加数学建模相关竞赛的大学生和指导教师,可以作为数学建模竞赛的辅导教程;也可供数学建模爱好者、应用数学和数学模型的教育工作者选用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有,侵权必究。

### 图书在版编目(CIP)数据

大学生数学建模竞赛指南 / 肖华勇主编. —北京: 电子工业出版社, 2015.4  
ISBN 978-7-121-25655-4

I. ①大… II. ①肖… III. ①数学模型—高等学校—教学参考资料 IV. ①O141.4

中国版本图书馆 CIP 数据核字(2015)第 046496 号

策划编辑: 祁玉芹

责任编辑: 鄂卫华

印 刷: 中国电影出版社印刷厂

装 订: 中国电影出版社印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 19.5 字数: 462 千字

版 次: 2015 年 4 月第 1 版

印 次: 2015 年 4 月第 1 次印刷

定 价: 45.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

服务热线:(010) 88258888。

# 前言

## PREFACE

近几年来，数学建模竞赛已成为大学生参与的热门竞赛。每年一届的全国大学生数学建模竞赛已成为全国高校规模最大的基础性学科竞赛，也是世界上规模最大的数学建模竞赛。大学生数学建模竞赛正以其独特的魅力吸引着各种专业、各种背景的老师和学生参与。

数学建模竞赛不仅是一项比赛，也是一种过程，一种理念，更是一种哲学。数学建模的主要目的是指导学生用建模的方法解决实际问题。在实际应用中，有些问题或许已经能够用已有的算法和公式来求解，但更多的问题没有固定答案，而且无法用简单的数学算法和公式来解决，因而，数学建模方法便是解决问题的有效途径。数学建模所培养的思维方式与技能，对于大学生在将来的工作中或读研究生涯中都会起到很大的帮助。对于工作或研究过程中的新问题，数学建模的思维方式与技能帮助学生快速地了解问题的背景，并由定性的分析转至定量的计算，从而给出比较有建设性的结论和指导性的意见。数学建模在企业的经营管理中有着举足轻重的地位，它在解决企业的运营成本、经济效益和社会效益等实际问题中发挥了重要作用；数学建模在对培养学生的科研能力方面有着其他专业课程无法替代的重要作用。在建立模型的学习过程中，学生需要查阅大量的文献资料、将实际问题抽象成数学模型，通过设计算法、模拟求解、撰写论文等，迅速提升学生们的实践能力，特别是做科学研究和撰写论文的能力。

本书通过对常用数学建模方法的讲解和实际问题的分析，培训学生思考、归纳、分析、创新的能力和技艺，同时也旨在帮助学生在大学生数学建模比赛中获得好成绩。

本书不只简单地介绍数学模型和案例，还全面的介绍数学建模和数学建模竞赛中的各个环节，包括团队组建时间安排、模型建立、程序实现以及论文写作，让参赛者对数学建模和竞赛有一个全面的认识，并能在比赛中做到运筹帷幄。

针对历年数学建模竞赛中常用的方法，本书不仅介绍方法，列举并分析比赛相关的案例，还给出并讲解实现的程序代码，让参赛者真正做到学以致用，而不是纸上谈兵。

本书还邀请了近几年国赛和美赛中的表现最优秀的参赛队来分析当年的赛题，讲述他们的模型和模型的程序实现，并分享比赛中的成功经验。让参赛的读者在全面了解国赛和美赛最优秀论文的同时，吸取成功者的经验。

本书还邀请了全国优秀指导教师，来分析平时数模教学活动以及比赛前后指导过程中



的经验，帮助年轻的数学建模老师更好地扮演指导教师这一角色。

本书由肖华勇担任主编，周吕文、赵松担任副主编。本书共分7章，第1章由谭欣欣、刚家泰和汪晓银编写；第2章由周吕文编写；第3章由周吕文、任立峰编写；第4章由肖华勇、周吕文编写；第5章由李文然、肖华勇、周吕文、熊风、舒毅潇编写；第6章由谭欣欣、肖华勇、周登岳、熊风编写；第7章由李勤伟、赵松、朱斌编写。林海卓与郭永蓉对本书的文字以及程序进行了校对。全书由周吕文、赵松统稿，肖华勇进行终审。

作为本书的总策划，校苑数模论坛（[www.mathor.com](http://www.mathor.com)）积极地协调了各方面资源，使得本书得以顺利地出版。在本书编写过程中，大连大学数学建模工作室的指导老师和学生（尤其是何玮、李祥、冯舒婷等同学），不仅参与了具体工作，还给予了我们很多支持和鼓励，在此表示感谢。另外本书的顺利编写和出版，还离不开谭忠、王钰聪、刘世尧、韩志斌、丁文超、李晶玲、董瑶、李蔓蔓、郑小娟、宋彦丽、徐平、邓赛、张哲、杨晓、张晶、师建鹏、李淑娟、刘思思、刘雅珊等人的支持，在此表示感谢。

感谢电子工业出版社的祁玉芹和韩婷两位老师，她们在本书的策划、选题的申报、写作大纲的确定以及编辑和出版等方面，付出了辛勤的劳动和智慧，给予我们很多支持和帮助。

书中可能有疏漏和不妥之处，欢迎大家批评指正，衷心希望广大读者与任课教师提出宝贵的意见和建议，以便再版时加以修正。读者可以发邮件到 [book@mathor.com](mailto:book@mathor.com) 与我们交流。

编者

2015年3月2日





# 目录

## CONTENTS

第 1 章 数学建模基本知识 .....	1
1.1 数学建模简介 .....	1
1.1.1 什么是数学建模 .....	1
1.1.2 初等数学模型案例 .....	3
1.1.3 数学建模的基本步骤与论文写作 .....	8
1.2 数学建模竞赛 .....	12
1.2.1 美国大学生数学建模竞赛 .....	12
1.2.2 中国大学生数学建模竞赛 .....	14
1.2.3 其他数学建模竞赛简介 .....	17
1.3 数学建模活动与能力培养 .....	22
1.3.1 数学建模与就业、升学、出国 .....	23
1.3.2 企业中的数学建模问题 .....	23
1.3.3 数学建模对科研和工作的影响 .....	24
第 2 章 数学建模中的团队合作 .....	25
2.1 组建团队 .....	25
2.1.1 程序员 .....	25
2.1.2 写手 .....	26
2.1.3 第三人 .....	27
2.1.4 团队合作 .....	27
2.1.5 赛前模拟 .....	28
2.2 时间安排 .....	29
2.2.1 赛前准备 .....	29
2.2.2 第一天：开始比赛 .....	29
2.2.3 第二天：建立模型 .....	29
2.2.4 第三天：写作和修改 .....	30



2.2.5	第四天：写作和润色	30
2.3	文献管理器	30
2.3.1	Zotero 的安装和配置	30
2.3.2	文献条目的保存	31
2.3.3	文献分类	31
2.3.4	文献引用	31
2.4	撰写论文	32
2.4.1	标题 (Title)	32
2.4.2	摘要 (Summary)	32
2.4.3	引言 (Introduction)	33
2.4.4	模型 (The Model)	33
2.4.5	解决方案 (The Solutions)	33
2.4.6	方案的比较 (Solution Comparison Methods)	34
2.4.7	结果 (Results)	34
2.4.8	结论-模型评价-改进方案 (Conclusions-S&W-FutureWork)	35
2.4.9	参考文献 (References)	35
2.4.10	论文的结构	35
2.5	论文排版	37
2.5.1	LaTeX 软件的安装和使用	37
2.5.2	简单示例	38
2.5.3	小节生成	38
2.5.4	公式输入	39
2.5.5	项目符号列表	39
2.5.6	图片插入	39
2.5.7	表格插入	40
2.5.8	引用文献	41
第 3 章	软件快速入门	43
3.1	MatLab 快速入门	43
3.1.1	引言	43
3.1.2	变量	44
3.1.3	矩阵和数组运算	45
3.1.4	控制结构语句	48
3.1.5	MatLab 文件	49
3.1.6	作图	51

3.2 Lingo 入门 .....	54
3.2.1 Lingo 基础知识讲解 .....	54
3.2.2 Lingo 实例 .....	60
<b>第 4 章 常用模型与算法 .....</b>	<b>71</b>
4.1 图论 .....	71
4.1.1 图论中 TSP 问题及 Lingo 求解技巧 .....	71
4.1.2 最短路算法及在建模中的应用 .....	79
4.1.3 状态转移与图论模型的巧妙结合 .....	92
4.1.4 最优树问题及 Lingo 求解 .....	96
4.1.5 竞赛图与循环比赛排名问题 .....	99
4.2 排队论模型 .....	106
4.2.1 排队论基本构成与指标 .....	107
4.2.2 排队论中四种重要模型 .....	108
4.2.3 排队论的计算机模拟 .....	118
4.3 元胞自动机简介及其在数学建模中的应用 .....	126
4.3.1 引言 .....	126
4.3.2 方法介绍 .....	128
4.3.3 应用举例 .....	133
4.4 启发式算法简介及其在数学建模中的应用 .....	142
4.4.1 引言 .....	142
4.4.2 模拟退火算法 .....	144
4.4.3 遗传算法 .....	150
<b>第 5 章 赛题解析 .....</b>	<b>157</b>
5.1 2013 CUMCM-A .....	157
5.1.1 问题综述 .....	157
5.1.2 分析与建模及求解 .....	157
5.1.3 论文点评 .....	183
5.2 2011 CUMCM-B .....	184
5.2.1 问题综述 .....	184
5.2.2 分析与建模及求解 .....	184
5.2.3 点评 .....	197
5.2.4 程序附件 .....	197
5.3 2014 MCM-A .....	209





5.3.1	引言	209
5.3.2	2014 MCM-A 题特等奖论文	217
5.3.3	程序实现	230
5.3.4	论文点评	232
5.4	2014 MCM-B	232
5.4.1	问题综述	232
5.4.2	分析与建模及求解	233
5.4.3	论文点评	244
5.5	2013 MCM-C	246
5.5.1	问题综述	247
5.5.2	分析与建模及求解	249
5.5.3	论文点评	262
<b>第 6 章</b>	<b>经验分享：做一名成功的指导者和参赛者</b>	<b>267</b>
6.1	优秀指导教师讲数模	267
6.1.1	数学建模活动是培养大学生创新能力的有效途径	267
6.1.2	谈谈我的数模路	272
6.2	获奖者的获奖之路	275
6.2.1	成长比成功更重要	275
6.2.2	如何准备美国大学生数学建模竞赛	283
6.3	本章小结	290
<b>第 7 章</b>	<b>互联网资源</b>	<b>291</b>
7.1	校苑数模论坛	291
7.1.1	历史介绍	291
7.1.2	校苑简介	292
7.1.3	展望	293
7.2	交叉学科数学建模学生论坛	293
7.2.1	创建论坛的初衷	293
7.2.2	交叉学科数学建模学生论坛五大亮点	296
7.2.3	总结与展望	296
	<b>参考文献</b>	<b>299</b>

# 第1章 数学建模基本知识

## 1.1 数学建模简介

### 1.1.1 什么是数学建模

提到数学，也许你的脑海里会浮出这样一副画面：鸦雀无声的教室，监考老师用警惕的目光扫视着全场，考生们分秒必争，疯狂地写下心中那一道道数学难题的答案，而“标准答案”其实早已被出题的专家们锁在了某个保险柜里。

那什么是“数学建模”？

“数学建模是指对现实世界的某一特定对象，为了特定的目的，做出一些重要的简化和假设，运用适当的数学工具得到一个数学结构，用它来解释特定现象的现实性态，预测对象的未来状况，提供处理对象的优化决策和控制，设计满足某种需要的产品等。”这里，我们用更为通俗易懂的话来定义数学建模——一项为生活服务的活动。

你玩过“人鬼过河”的游戏吗？三个人和三个鬼要过河，只有一条船，船上最多可以乘两个人或两个鬼或一人一鬼，但河岸上鬼的数量不能大于人的数量，否则人会被鬼所吞噬。那么怎样合理设计过河路线才能保证这三个人安全渡到河的对岸呢？显然这是一个锻炼人的逻辑思维的游戏，也许你会一遍遍的尝试，寻找合理的过河方法。而它，从逻辑思维角度分析就是一道数学建模题目。因此我们说数学建模可以是生活中的智力游戏，让我们的生活更加丰富多彩。

你喜欢旅游吗？你想把全中国的各个省市的名胜景点都走一遍吗？那么怎样设计一条旅行路线才能让我们的行程最短，所需费用最少呢？或许你会打开百度地图，一遍遍地计算，寻找最短行程。但是走进数学建模的世界，你会发现只需要在电脑上敲出几行代码，做一个小程序就可以知道最短距离是 15651.8km。这就是数学建模里面著名的“TSP”问题。显然，我们也可以说数学建模是帮助我们解决生活中的小问题，让我们更好的享受生活。

你们班有 60 人，现有一个出国留学的名额，那么你能够拥有这个机会的可能性有多少？也许你会不假思索的给出答案：1/60。也许你的答案是正确的，但是从数学建模的角度分析，你的答案也许就不是那么有说服力了。因为你忽略了事情的前提条件：你有没有想过也许会存在潜规则和黑幕呢？如果这样，那么哪里还会有 1/60 的概率落在你的头上。再如果要考虑到学生的家庭经济状况及学生的性别、年龄等诸多主要因素，你出国留学的概率又会是多少呢？显然，数学建模可以帮助我们解决学习或工作中的问题。





讲述了这三个生活中常见的小事，不知你对建模是否有了更进一步的了解。从理论上讲，数学建模，虽名曰数学，但又与纯数学竞赛有着天壤之别。它既不是纯粹的数学竞赛，也不是纯粹的计算机竞赛，而是涉及多学科、多领域，考查学生处理实际问题的综合能力。它不像考试，更像是一个课题小组在规定的时间内突击完成一项任务。

郑州大学的石东洋教授解释道：“数学建模就是以各学科知识为基础，利用计算机和网络等为工具，来解决实际问题的一种智力活动。它既不是传统的解题，也不同于其他赛事，而是更重视应用与创新以及动手能力的考察。”

随着社会的发展，数学在社会各领域中的应用越来越广泛，不仅运用于自然科学的各个领域，而且渗透到经济、军事、管理以至于社会活动的各领域。但社会对数学的需求并不只是需要专门从事数学研究的人才，而是需要在各部门中从事实际工作的人善于运用数学的思维方法来解决他们每天面临的大量的实际问题，取得更高的经济效益。对于生活中复杂的实际问题，发现其内部规律，用数学语言将其描述出来，进而把这个复杂的实际问题转化为一个简化的数学问题，这就是数学模型，建立数学模型的过程其实就是数学建模。当然，复杂的实际问题中有许多因素，在建立模型中是不可能毫无遗漏的将其全部考虑在内，只考虑其中最主要的因素将数学模型建立起来就可以了，这样就可以用数学工具和数学方法去解答生活工作中一个实际问题。

那么你见过数学建模竞赛的场面是什么样的呢？它也和常规的数学竞赛一样两个小时一张试卷吗？当然不是。有人这样描述，全国乃至世界范围内的大学生，来自不同学院、不同专业的建模爱好者们，三人一队，一起参加历时三天三夜或四天四夜的建模比赛。他们有的在娴熟地敲打着键盘、点击着鼠标，聚精会神地凝视着电脑屏幕上的一篇篇文献；有的两眼紧紧盯着那屏幕上来回滚动的数字和符号，仿佛在看武侠小说、侦探片、世界杯；有的则在那堆积如山的建模书里翻来覆去地搜索着。每位建模者都有对赛题独特的观点和见解，他们彼此交流，只为了找到自己建模思路中的某个“元件”，从而完善自己的建模大厦。当然，数学建模竞赛并没有一个固定的答案，完成数学建模赛题关键在于团队自己的创新能力。而人的创造力是没有顶峰的，每个团队都应竭尽全力，没有最好，只有更好。因此每年全国评出的优秀答卷几乎都有不足之处，这并不奇怪，因为答卷的优秀与否是相对而言的。

数学建模的益处当然不仅仅在于比赛的过程使人增长知识，开阔视野，而且对我们日后的学习或是工作也有很大帮助。中国科学院攻读空间物理博士学位的一位建模爱好者说：“我目前的工作就是分析卫星数据，从中抽取相关物理规律。这是个非常麻烦、繁琐的过程，并且还需要自己学习一些计算机语言，编程序，看大量英文文献，还要和导师以及一些专家合作讨论。可以说，在数学建模活动中锻炼的这几年，让我对目前这些困难的解决能够应付自如。”

毕业后走入工作岗位的一位建模爱好者这样描述：“目前我在一家大型电子商务公司做平台运营，负责七个店铺在四个平台中的日常销售。电子商务中无数的数据之间相互影响、相互依托，让我更乐于用建模的思维去思考因子之间的相关性，客户的行为分析，地

域分析, 访问量, 浏览量, 转化率对成交金额的影响。提升店铺 DSR 评分, 提高转化率, 促进成交金额, 使我在平凡的工作中表现的更加自信, 在玩转数据之间更加从容。”

21 世纪以来, 人类已经进入到了以计算机、网络、数码、光纤、多媒体为主要标志的信息时代, 定量化、数字化的技术得到了飞速发展, 并应用于一切领域, 培养应用型数字人才已迫在眉睫。数学建模, 不仅丰富了我们的课余生活, 丰富了我们的大脑, 开拓了我们的视野, 让我们与全国乃至世界的大学生一起站在同一个平台上角逐, 更为我们以后顺利走入工作岗位奠定了基础。

现在, 你该知道什么是数学模型和数学建模了吧! 从错综复杂的实际问题中, 经过合理的分析、假设, 抓住主要矛盾、忽略次要矛盾, 得到一个用数学的符号和语言的表达式, 就是数学模型。综合运用所学知识, 选择适当的方法加以解决就是数学模型的求解。这种从实际中提出问题、建立数学模型到模型求解的完整过程就是数学建模。

### 1.1.2 初等数学模型案例

数学模型是将现象加以归纳、抽象的产物, 它源于现实, 又高于现实; 只有当数学建模的结果经受住现实对象的检验时, 才可以用来指导实际, 完成实践——理论——实践这一环节。

现实世界中有很多问题, 它的机理比较简单, 一般用静态、现态、确定性模型描述就能达到建模的目的, 基本上可以用初等数学模型的方法来构造和求解模型。

初等数学模型中的大多数问题都是很早就提出来了, 这些问题简直像天方夜谭似的极其有趣, 表面上看无从下手。而数学建模则是将原型进行适当的简化、提炼而构成的一种原型代替物。这种代替物并不是原型原封不动的复制品。原型有各个方面和各种层次的特征, 模型只反映了与某种目的有关的那些方面和层次的特征, 从而达到解决某个具体问题的目的。

**例 1:** 人、猫、鸟、米均要过河, 船上除 1 人划船外, 最多还能运载 1 物, 而人不在场时, 猫要吃鸟, 鸟要吃米, 问人、猫、鸟、米应如何过河?

**模型假设:**

人、猫、鸟、米要从河的南岸到河的北岸, 由题意, 在过河的过程中, 两岸的状态要满足一定条件, 所以该问题为有条件的状态转移问题。

**模型建立:**

我们用  $(w, x, y, z)$ ,  $w, x, y, z=0$  或  $1$ , 表示南岸的状态, 例如  $(1, 1, 1, 1)$  表示它们都在南岸,  $(0, 1, 1, 0)$  表示猫, 鸟在南岸, 人, 米在北岸; 很显然有些状态是允许的, 有些状态是不允许的, 用穷举法可列出全部 10 个允许状态向量,  $(1, 1, 1, 1)(1, 1, 1, 0)(1, 1, 0, 1)(1, 0, 1, 1)(1, 0, 1, 0)(0, 0, 0, 0)(0, 0, 0, 1)(0, 0, 1, 0)(0, 1, 0, 0)(0, 1, 0, 1)$ 。



**模型求解:**

将 10 个允许状态用 10 个点表示, 并且仅当某个允许状态经过一个允许决策仍为允许状态, 则这两个允许状态间存在连线, 而构成一个图, 如图 1-1 所示, 在其中寻找一条从  $(1,1,1,1)$  到  $(0,0,0,0)$  的路径, 这样的路径就是一个解, 可得下述路径图。

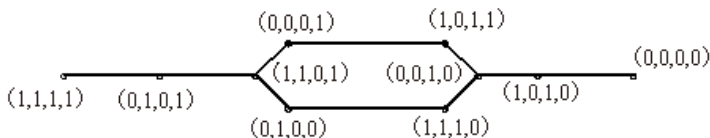


图 1-1

由图 1-1, 有两个解都是经过 7 次运算完成, 均为最优解。

**模型推广:**

这里讲述的是一种规格化的方法, 所建立的多步决策模型可以用计算机求解, 从而具有推广的意义, 适当地设置状态和决策, 确定状态转移律, 建立多步决策模型, 是有效地解决很广泛的一类问题的方法。

**例 2:** 某新婚夫妇急需一套属于自己的住房。他们看到一则理想的房产广告: “名流花园之高尚住宅公寓, 供工薪阶层选择。一次性付款优惠价 40.2 万元。若不能一次性付款也没关系, 只付首期款为 15 万元, 其余每月 1977.04 元等额偿还, 15 年还清 (公积金贷款月利息为 3.675‰)。贷款额为多少?”

**模型假设:**

贷款期限内利率不变; 银行利息按复利计算。

**符号定义:**

$A$  (元): 贷款额 (本金);  $n$  (月): 贷款期限;  $r$ : 月利率;  $B$  (元): 月均还款额;  $C_k$ : 第  $k$  个月还款后的欠款。

**模型建立:**

将该递推数列变形为:

$$C_k - \frac{B}{r} = (1+r) \left( C_{k-1} - \frac{B}{r} \right) \quad k = 1, 2, \dots, n$$

利用等比数列得到一般项公式为:

$$C_n - \frac{B}{r} = \left( C_0 - \frac{B}{r} \right) (1+r)^n$$

由  $C_n = 0$  有:

$$A = \frac{B}{r} - \frac{B}{r} (1+r)^{-n} = \frac{B}{r} \cdot \frac{(1+r)^n - 1}{(1+r)^n}$$



模型求解：

带入： $n=180$ 、 $r=0.003675$ 、 $B=1977.04$

则： $A=260000$ （元）。（因每月还款 1977.04 只能精确到分，实际计算结果为 259999.4 元）

例 3：世界纪录的赛跑数据如表 1-1 所示。

表 1-1

距离 $x$ 米	100	200	400	800	1000	1500
时间 $t$	9.95"	19.72"	43.86"	1'42.4"	2'13.9"	3'32.1"

研究运动员跑过的距离长度是怎么影响其成绩的？

模型假设：

运动员的成绩仅与跑过的距离长度相关，即不考虑运动员的自身差异及场地，环境等差异的影响。

模型建立：

在坐标系上将数据对应的点一一标出来，如图 1-2 所示，这些点基本上分布在一条直线附近，猜想两者之间有线性关系。

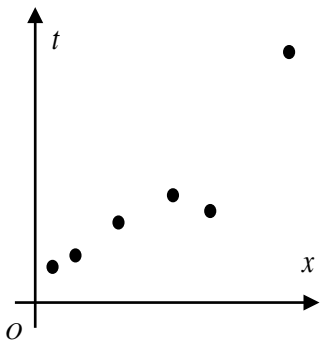


图 1-2

模型修正：

由于数据点并不严格在一条线上，设想其误差由长度以外的其他因素所导致，因此，模型修改为

$$t_i = f(x_i) + \varepsilon_i = a + bx_i + \varepsilon_i$$

模型求解：

利用二元函数最小值的方法，不难求得

$$b = \frac{\sum_{i=1}^n x_i t_i - n \bar{x} \bar{t}}{\sum_{i=1}^n x_i^2 - n \bar{x}^2} = 0.1455$$





$$a = \bar{t} - b\bar{x} = -9.99$$

$$t = -9.99 + 0.1455x$$

#### 例 4: 投掷铅球的最佳角度问题

用数学方法研究体育运动是从 20 世纪 70 年代开始的。1973 年, 美国的应用数学家 J·B·开勒发表了赛跑的理论, 并用他的理论训练中长跑运动员, 取得了很好的成绩。几乎同时, 美国的计算专家艾斯特运用数学和力学, 并借助计算机研究了当时铁饼投掷世界冠军的投掷技术, 从而提出了他自己的研究理论, 据此改进了投掷技术的训练措施, 并使这位世界冠军在短期内将成绩提高了 4m。这些都说明了数学在体育训练中发挥着越来越明显的作用。

在铅球投掷训练中, 教练关心的核心问题是投掷距离。而距离的远近主要取决于两个因素: 速度和角度。在这两个因素中, 哪个更为重要呢?

#### 模型假设:

铅球投掷训练涉及的变量很多, 为简化问题, 我们在下面的模型中, 将不考虑铅球运动员在投掷区域内身体的转动, 只考虑铅球的出手速度与投射角度这两个因素。并作如下假设:

- (1) 忽略铅球在运行过程中的空气阻力作用;
- (2) 投射角度与投射初速度是相互独立的两个量;
- (3) 将铅球视为一个质点。

#### 模型建立:

先考虑铅球从地平面以初速度  $v$  和角度  $\theta$  投掷出的情形。如图 1-3 所示, 铅球在点  $P$  处落地。

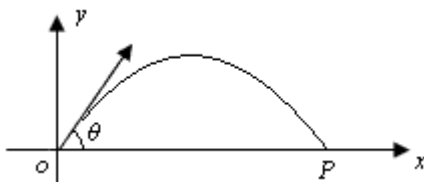


图 1-3

先来求铅球的运动方程。

设铅球在时刻  $t$  的动点坐标为  $(x, y)$ , 得运动方程

$$\begin{cases} x = v \cos \theta \cdot t \\ y = v \sin \theta \cdot t - \frac{1}{2} g t^2 \end{cases}$$

消去方程中的参变量  $t$ , 得到关于  $x, y$  的关系式

$$y = -\frac{g}{2v^2 \cos^2 \theta} x^2 + \tan \theta \cdot x$$

为了求出铅球落地处的坐标, 只需令  $y=0$ , 解得

$$x_1 = 0, \quad x_2 = \frac{2v^2 \sin \theta \cos \theta}{g} = \frac{v^2 \sin 2\theta}{g}$$

其中  $x_1$  是铅球起点的坐标,  $x_2$  是铅球落地时点  $P$  的坐标。

若  $v$  固定, 则投掷距离是投射角  $\theta$  的函数。当  $\theta = 45^\circ$  时, 投掷距离达到最大值, 这时的投掷距离为  $\frac{v^2}{g}$ 。这就是说, 按  $45^\circ$  角投掷时, 投掷的距离最远。

然而, 上述模型与实际是有差距的。这是因为, 铅球不是从地面上出手, 而是从一定的高度处出手。因而上面的方程应调整为:

$$\begin{cases} x = v \cos \theta \cdot t \\ y = v \sin \theta \cdot t - \frac{1}{2} g t^2 + h \end{cases}$$

消去  $t$ , 得到:

$$y = -\frac{g}{2v^2 \cos^2 \theta} x^2 + \tan \theta \cdot x + h$$

令  $y=0$ , 得方程:

$$-\frac{g}{2v^2 \cos^2 \theta} x^2 + \tan \theta \cdot x + h = 0$$

解之得:

$$x_{1,2} = \frac{v^2 \sin 2\theta}{2g} \pm \sqrt{\left(\frac{v^2 \sin 2\theta}{2g}\right)^2 + \frac{2v^2 h \cos^2 \theta}{g}}$$

舍去负根, 得到点  $P$  的坐标为:

$$x = \frac{v^2 \sin 2\theta}{2g} + \sqrt{\left(\frac{v^2 \sin 2\theta}{2g}\right)^2 + \frac{2v^2 h \cos^2 \theta}{g}}$$

即铅球的射程为:

$$\frac{v^2 \sin 2\theta}{2g} + \sqrt{\left(\frac{v^2 \sin 2\theta}{2g}\right)^2 + \frac{2v^2 h \cos^2 \theta}{g}}$$

数值模拟:

取  $g=10 \text{ m/s}^2$ ,  $h=1.6 \text{ m}$ , 利用这一公式, 列表给出速度与角度对投掷距离的影响, 如



表 1-2 所示。

表 1-2

速度 $v$ (m/s)	角度 $\alpha$	距离 $x$ (m)
11.5	47.5	14.929
11.5	45.0	15.103
11.5	42.5	15.182
11.5	41.6	15.189
11.5	40.0	15.169
11.5	38.0	15.092
11.5	36.0	14.960
11.0	41.6	14.032
12.0	41.6	16.395

从上表可以看出，当  $v=11.5\text{m/s}$  时，最佳角度为  $41.6^\circ$ （可用微积分知识得到）。当角度在  $38^\circ$  到  $45^\circ$  间变化时，产生的距离差是  $0.097\text{m}$ ，角度  $\frac{45-38}{38} \approx 16\%$  的偏差引起距离  $0.06\%$  的偏差。速度从  $11\text{m/s}$  变到  $12\text{m/s}$  引起了距离从  $14.032\text{m}$  到  $16.359\text{m}$  的偏差，也就是说，速度  $9\%$  的增加导致了距离  $16.8\%$  增加。这个结果表明，教练在训练运动员时，应集中主要精力来增加投掷的初始速度。

模型评价：

- （1）上面的模型比较粗糙，还有许多因素没有考虑到，例如运动员的身体转动，投掷者的手臂长度，肌肉的爆发力、铅球的质量，等等。加上以上诸因素后，得出的公式自然会更精确，但处理起来会复杂得多。
- （2）关于速度与角度的偏差百分率的计算，是否可以比较还值得商榷。
- （3）铅球投掷问题的数学模型，可以应用于铁饼、标枪或篮球投篮等投掷问题，读者不妨用类似上面的方法进行研究。

当实际问题需要我们对所研究的现实对象提供分析、预报、决策、控制等方面的定量结果时，往往都离不开数学的应用，而建立数学模型则是这个过程的关键环节。

### 1.1.3 数学建模的基本步骤与论文写作

#### 一、数学建模的基本步骤

通过以上几个例子，我们发现建立数学模型的基本步骤就是解决一个实际问题的基本步骤。由于实际问题的背景、性质、建模的目的等方面不同，因此，建模要经过哪些步骤并没有固定的模式和标准。数学建模的基本步骤包括以下 7 个主要部分：

##### 1. 模型准备及问题分析

当看到竞赛题目时，首先，需要对问题剖析，抓住问题本质和主要因素，确定问题的关键词，进行查阅资料和文献，了解问题的实际背景、相关数据或相关研究进展情况，获

得关键资料,并初步确定研究问题的类型。竞赛的问题都是来自于实际生活中的各个领域,并没有固定的方法和标准的答案,所以,要明确问题中所给的信息点,把握好解决问题的方向和目的,仔细分析问题关键词和数据信息,可适当补充一些相关信息和数据(具有一定权威性)。为接下来的模型建立奠定基础和基本方向。

### 2. 模型的假设

竞赛题目都是来自于实际生活,所涉及的方面较广、受影响的因素较多,而在建模过程中不可能面面俱到,故需结合问题的实际意义,适当地将一些因素简化,但必须不能对问题主要因素影响太大。抓住问题关键、忽略次要因素,进行合理化的简要假设,这是为建模过程中排除一些较为难处理的情况,使建立的模型更趋优化和合理,也是评价一个模型优劣的重要条件。

### 3. 模型的建立

通过所做的分析和假设,结合相关的数学基本原理和理论知识,将实际问题转化为数学模型,可以用数学语言、符号进行描述和表示问题的内在现象和规律。结合相关学科的专门知识,根据所提供的要求和信息,建立一个关于问题中主要变量与主要因素间的数学规律模型,可以以数学方程式、图形、表格、数据和算法程序等形式表示。但在建模过程中应多创新,不要一味效仿,可以将多个知识点进行穿插和结合,如基于 K-means 的粒子群改进算法。还可以在算法程序上进行改进和优化,体现模型的创新性。

### 4. 模型的求解

在模型求解过程,会用到传统的数学方法,如解方程、公式证明、统计分析等,但目前,最广泛使用的是数学软件和计算机技术,如 MatLab、Lingo、SPSS 等。有时还需要掌握一门编程语言,或新的软件,所以需要具备针对实际问题学习新知识的能力,能够灵活应用新知识并将其与实际问题结合以对模型求解。

### 5. 结果分析与检验

对所求的结果,针对问题的实际情况和意义进行分析。可以通过误差分析、灵敏度分析,来表现模型解决实际问题的效果及实际应用的范围。误差分析,可以适当地调整模型,或提出出现误差的可能原因或解决的方案;灵敏度分析,是针对某些主要参数的,可以确定模型中主要变量和参数的误差允许范围。有时需要通过将所得数据进行方差、标准差、 $t$  检验或  $f$  检验等。通过分析和检验,充分表现模型的合理性和可行性。

### 6. 论文写作

数学建模比赛,不仅需要利用各种数学、物理、智能算法等来解决问题,还需要将研究成果撰写成论文,以电子版形式上交。故按照数学建模的基本步骤,建立一个恰当的数学模型并求解,使参赛者清晰明了地表达解题思路,以展示自己能力,也是评委评定一篇论文好坏的唯一依据。所以完成一篇高质量的竞赛论文不仅能展示自我才能,也能为





竞赛加分。

## 7. 模型的应用

以上是将实际问题转化为数学模型进行求解并证明，在进行大量研究和演绎后，最终还需将其回归到实际，看其是否具有合理性和可行性。这需要用实际信息或数据进行验证。

以下为三种数学建模基本步骤，可根据个人所需对各个部分进行调节，如表 1-3 所列。

表 1-3

一	二	三
1. 摘要	1. 摘要	1. 摘要
2. 问题重述	2. 问题的提出与重述、问题的分析	2. 问题的叙述、背景的分析
3. 问题的分析	3. 变量假设	3. 模型的假设、符号说明
4. 模型假设	4. 模型建立	4. 模型建立
5. 符号说明	5. 模型求解	5. 模型求解
6. 模型建立	6. 模型分析与检验	6. 模型检验
7. 模型求解	7. 模型的评价与推广	7. 模型评价
8. 结果分析、验证、模型检验及修正	8. 参考文献	8. 参考文献
9. 模型评价	9. 附录	9. 附录
10. 参考文献		
11. 附录		

## 二、数学建模的论文写作

下面按照第一种数学建模基本步骤，就论文写作部分进行详细叙述。

### 1. 摘要

摘要就是一篇建模比赛论文的整体面貌，评委对论文第一轮评审就是通过对摘要进行筛选，所以对于每个参赛队来说，写好摘要，是获奖的较为重要的一步，也是为论文进一步的得到评委审批的关键。

摘要的字数一般在 400~800 字，但其内容却包含了参赛队对题意理解、模型类型、建模思路、采用的求解方法及求解思路、算法特点、灵敏度分析、模型检验、主要数值结果和结论等。

在摘要下一行还需选取 3~5 个关键词，用来彰显竞赛论文的主要内容。

### 2. 问题重述（或问题的提出与重述）

通过自己对题意的理解，用自己的语言重新描述问题。如果问题本身很简短，可以抄题，一般情况下不建议抄题。有些情况，可以结合问题的背景进行简要扼明说明解决问题的意义所在。

### 3. 问题分析

需要抓住题目的关键词和主要目地及要求，分析要中肯、确切。依据原理要明确，描

述要简明扼要，可列出关键步骤，切记不要冗长，繁琐。对问题的分析，可以作为第三部分，也可以将其针对每个问题写在模型建立中。建议采用流程图，使思路表述更清晰。

### 4. 模型假设

在对问题进行分析后，发现有些因素或条件，还无法进行考虑或估算；或是针对问题的主要因素，舍弃次要因素的影响，采用假设的方式，是我们解决的问题简化，模型更合理化。这部分内容，可以单独写，也可以在模型建立时根据所需要情况再进行描述。

### 5. 符号说明

对模型使用的变量加以说明，以简要的文字表述各字母的意义，其中各个主要符号的大小写、英语和阿拉伯语，要与正文中符号一致。符号说明太多时，建议采用表格形式。有时可将其分布在模型建立中，也可以用表格形式。

### 6. 模型的建立

明确题意后，简述基本思路。首先，简要介绍利用的基本原理和基本思想，再进行构建基本模型，如数学表达式、构建方案、构造图、算法流程图等，要明确说明解题的思想和思路，有逻辑性、合理性、可行性，完整叙述。结合实际问题，进行改进和完善基本模型，使其能有效、实用解决问题。

### 7. 模型的求解

采用蚁群算法、模拟退火、遗传算法、元胞自动机、蒙特卡洛等一些智能算法时，要简要写明算法步骤，要阐明使用理由。计算时将一些必要的步骤列出来，不用将中间的计算过程一一列出。

### 8. 模型的检验

在模型求解后，采用一些方法进行检验。有的可以采用原始数据和查找的数据处理效果进行对比检验；有的采用对结果的 $t$ 检验、 $f$ 检验等，若误差较大时，可分析原理，进行改进或修正。

### 9. 模型评价和推广

这里需要强调的是，衡量一个模型的优劣全在于它的应用效果，而不是采用了多么高深的数学方法。进一步说，如果对于某个实际问题，我们用初等的方法和所谓的高等方法建立了两个模型，它们的应用效果相差无几，那么受到人们欢迎并采用的一定是前者而非后者。

模型推广，可以采用将原题要求进行扩展，进一步讨论模型的实用性和可行性；还可以提出问题的展望。

### 10. 参考文献

论文提及或是直接引用的文献、引用数据的出处等，需要在这部分进行罗列出来。常







用的文献表述形式如下[1]:

(1) 公开发表的杂志

[序号] 作者, 文章名字 [文献类型], 刊物名, 出版年, 出版单位, 卷号 (期号), 起止页码。

如: [5] 李海芳, 杨红云, 张英等. 四氧化三铁/单壁碳纳米管磁性复合纳米粒子分散固相微萃取-高效液相色谱法测定牛奶中的香精添加剂 [J]. 色谱, 2014,(4)413~418.

(2) 公开出版的书籍

[序号] 作者, 书名 [M], 版次, 出版地, 出版单位, 出版年; 起止页码。

如: [3] 唐焕文, 贺明峰. 数学模型引论 [M]. 北京: 高等教育出版社, 2001.

(3) 网页资料类

[序号] 作者, 资源标题, 网址, 访问时间 (年月日)。

如: 能斯特方程, <http://baike.baidu.com/view/404720.htm?fromtitle=能斯特方程式&fromid=1214555&type=syn>, 2014-11-28.

英文写作也是这样的要求。一般我们可以采用上述格式。

其中的参考文献类型标识字母有: J—期刊、M—专著、N—报纸、C—论文集、D—学士论文、P—专利、R—报告、S—标准。

## 11. 附录

这部分不属于论文的正文内容, 是一些很重要的计算过程、算法程序、一些数据表格等。

古训有云: 读万卷书, 行万里路。一个优秀的学习者并不仅仅是掌握理论上的知识, 更应将所学的知识应用到实际中, 不断的在实践中提升自我。这也是素质教育对于同学们的要求。

## 1.2 数学建模竞赛

### 1.2.1 美国大学生数学建模竞赛

纵观历史, 我们不难发现, 任何一项成熟、成功的技术一定会进入培养人才的教育领域。因此数学教学必须自觉贯彻素质教育的精神, 使同学们不仅学到许多重要的数学概念, 方法和结论, 而且领会到数学的精神实质和思想方法, 掌握数学这门学科的精髓, 使数学成为他们手中得心应手的武器, 终身受用不尽。

正是由于认识到培养应用型数学人才的重要性, 而传统的数学竞赛不能担当这个任务。在美国就有一些有识之士开始探讨组织一项应用数学方面的竞赛可能性。经过论证、争论, 终于在 1985 年举办了第一届美国大学生数学建模竞赛。

美国大学生数学建模竞赛分为 MCM (Mathematical Contest in Modeling) 和 ICM





(Interdisciplinary Contest in Modeling) 两种。MCM 始于 1985 年, 由美国数学及其应用联合会 (COMAP) 和美国国家安全局 (NSA) 联合主办。2013 年起, 由 COMAP 主办, 美国工业与应用数学学会 (SIAM)、美国运筹学和管理科学学会 (INAFORMS)、美国数学协会 (MAA) 及中国工业与应用数学学会 (CSIAM) 协办。MCM 为专业性较强的数学建模竞赛, 包括 A、B 两道题目。1999 年 COMAP 推出了交叉学科建模竞赛 (ICM), 赛题在原来 MCM 的 A、B 两题基础上, 增加了 C 题 (ICM)。竞赛的宗旨是鼓励大学生运用所学的知识 (包括数学知识及其各方面的知识) 去参与解决实际问题。这些实际问题并不限于某个特定领域, 可以涉及非常广泛的、并不固定的范围。对数学知识要求不深, 一般没有事先设定的标准答案, 但留有充分余地供参赛者发挥其聪明才智和创造精神, 来促进应用型人才的培养。

MCM/ICM 每年举办一届, 在每年的二月份举行, 历时 4 天 4 夜 96 小时, 由三人组成一队, 每队有一名指导老师。MCM/ICM 面向全球的大学生, 参赛队伍通过网络报名, 每队交纳 100 美元报名费, 缴费成功后每队将生成唯一的参赛队号。在参赛期间, 队员们可以充分发挥想象力, 查阅任何图书或互联网上的资料, 利用计算机、软件等外部资源, 就指定的问题完成从建立模型、求解、验证到论文撰写的全部工作。竞赛中唯一的禁律, 就是在竞赛期间不得与队外任何人 (包括指导教师) 讨论赛题。

竞赛结果以论文的形式交付, 美国大学生数学建模竞赛没有事先设定的标准答案, 评卷的标准为: 假设的合理性, 建模的创造性, 结果的正确性以及论文表述是否清晰。专家在评卷时并不对论文给出分数也不采用通过、失败这种记分, 而只是将论文分成一些级别: Outstanding (特等奖), Meritorious (一等奖), Honorably mention (二等奖), Successful participation (成功参赛奖)。其中, 绝大多数队伍能够获得成功参赛以上的奖项, 一等奖、二等奖、成功参赛奖的比例控制在 15%, 30%, 50% 左右, 而特等奖及特等奖提名奖 (2010 年起设立) 的评选相当严格, 获奖队伍也相对较少, 特等奖一般不超过 20 个队。表 1-4 所示是美赛近十年来总参赛队数及各个等级的获奖比例统计。

表 1-4

年 份	参赛队数	特 等 奖	提 名 奖	一 等 奖	二 等 奖	成功参赛	获奖比例
2005	828	13	—	13.4%	34.3%	50.7%	98.5%
2006	972	16	—	17.5%	31.0%	49.9%	98.4%
2007	1222	16	—	13.5%	34.7%	50.4%	98.6%
2008	1542	12	—	13.9%	42.3%	42.9%	99.1%
2009	2049	11	—	16.1%	21.6%	61.8%	99.5%
2010	2610	13	18	18.2%	25.4%	54.6%	98.2%
2011	3528	14	28	14.2%	32.1%	51.9%	98.2%
2012	5026	17	21	10.6%	33.6%	55.0%	99.1%
2013	6593	16	20	14.8%	30.9%	53.7%	99.4%
2014	7768	19	17	10.1%	32.6%	56.8%	99.5%

美国大学生数学建模竞赛的开展, 有利于学生知识、能力和素质的全面培养, 既丰富、活跃了广大同学的课外活动, 也为优秀学生脱颖而出创造了条件。每个参加 MCM/ICM 并





获得成功参赛以上奖项的队伍及其指导老师都将获得一张证书，部分特等奖论文将于同年刊登在美国著名的数学杂志 UMAP 上。近年来，一些特等奖获奖学生还将获得现金奖励并被邀请参加专业学会的年会作报告，不少大学愿意提供奖学金给最优秀的那些队员去该校读应用数学方面的研究生。第一届 MCM 时，就有美国 70 所大学 90 个队参加。到 1992 年，仅仅 7 年，已有包括美国的大学在内的 189 所大学的 292 个队伍参加。在某种意义上，美国大学生数学建模竞赛已经成为了一项国际性的竞赛。近三十年来吸引了大量世界著名高校参赛，其中不乏哈佛大学、普林斯顿大学、麻省理工大学、清华大学等国际名校。

随着美国大学生数学建模竞赛的影响力越来越大，得到越来越多的学校和单位认可。1989 年，我国大学生首次参加美国大学生数学建模竞赛。到 1996 年，我国已经有 39 所大学 115 个队参加，复旦大学和中国科技大学首次取得两项特等奖。特别是 2009 年，COMAP 取消了每个学校或机构只允许参加 7 支队伍的限制 [2009 年以前每个学校（机构）MCM 只能参加 4 支队伍，ICM 只能参加 3 支队伍]，来自全世界的参赛队伍增长速度加快，同年我国共有 1624 个队伍参加竞赛。2014 年，我国的参赛队数也从最初的 4 个队增加至 7273 个队，其中包括来自清华大学、北京大学、浙江大学、重庆大学等知名高校的学生参与此项赛事。我国参赛队伍在竞赛中所占比例越来越大，而且呈逐年增长的趋势，近十年（2005—2014）来我国的具体参赛情况如图 1-4 所示。

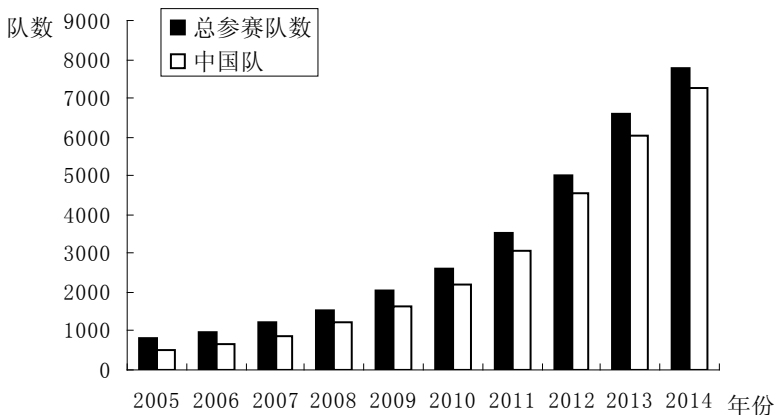


图 1-4

正是因为美国大学生数学建模竞赛在培养学生创新能力，提高学生实践技能，拓展学生知识面的突出作用，1994 年起，我国拥有了自己的大学生数学建模竞赛。

### 1.2.2 中国大学生数学建模竞赛

全国大学生数学建模竞赛起源于美国大学生数学建模竞赛。1985 年，首届数学建模竞赛在美国举办。1988 年 6 月，北京理工大学叶其孝教授在美国讲学期间向美国大学生数学建模竞赛发起者和负责人 Fusaro 教授了解这项竞赛的情况，商讨中国学生参赛的办法和规则。1989 年，我国大学生（来自清华大学、北京大学、北京理工大学 3 所学校的 4 个队）

首次参加美国大学生数学建模竞赛。经过一段时间的参与之后，老师和学生普遍认为数学建模竞赛能够使大学生多方面的能力得到锻炼，比如提高了大学生理论联系实际及思考问题的能力，锻炼了大学生从互联网上查阅文献、收集资料的能力，提高他们的文字表达水平，培养了他们同舟共济的团队精神和进行协调的组织能力，等等。而这些能力的锻炼会使他们终生受益。自从1989年以来，我国越来越多的大学生参与了美国大学生数学建模竞赛，历年来都取得了较好的成绩。

1990年6月，美国Fusaro教授访问北京和上海，作了有关美国大学生数学建模竞赛的报告，并与叶其孝、姜启源教授等讨论了中国数学建模竞赛的组织工作。为了使更多的大学生受益，1992年，由中国工业与应用数学学会组织了我国首届大学生数学建模联赛，教育部领导及时发现并扶植培育了这一新生事物，决定从1994年起由教育部高教司和中国工业与应用数学学会共同举办全国大学生数学建模竞赛，每年一届。

1993年，全国大学生数学建模竞赛仅有16省市、101所院校的429队参加，从此以后，每年参与竞赛的学校和参赛队伍呈现快速增长的趋势。2002年7月，全国组委会与我国最大的出版社之一的高等教育出版社签订协议并获得赞助，将这场竞赛正式命名为“高教社杯全国大学生数学建模竞赛”。2010年9月，全国除台湾省外的所有33省（市、自治区，包括港澳），以及来自新加坡和澳大利亚共1197所院校参加，这是首次有外国大学生参加本项竞赛。对于来自中国大陆以外的参赛者，该竞赛首次被称为当代大学生数学建模竞赛（Contemporary Undergraduate Mathematical Contest in Modeling，简称为CUMCM）。如图1-5和图1-6所示分别为从1993年到2014年参赛院校总数和参赛总队数随年份的变化图。

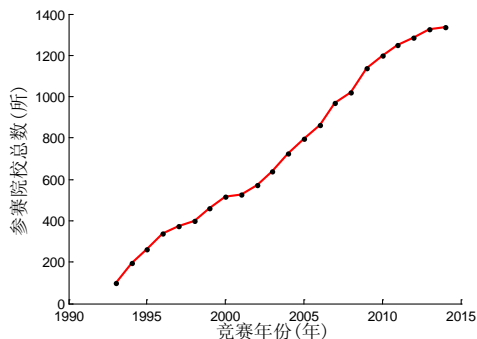


图 1-5

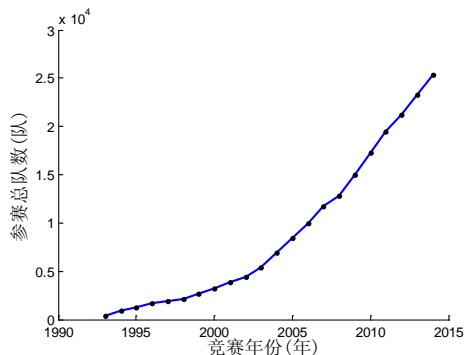


图 1-6

从图中可以发现，两条曲线都呈急剧上升趋势，其中参赛院校增长率从2010年起渐趋缓慢，且院校总数已达1300之多，说明参赛院校已接近饱和，该比赛已经在高校普及；而参赛总队数的年增长率却逐渐增大，说明越来越多的大学生积极的参与进来。从1993年16省市101所院校的429队，到2014年来自全国33个省/市/自治区（包括香港和澳门特区）及新加坡、美国的1338所院校、25347个队（其中本科组22233队、专科组3114队），该比赛已经走过23个年头，它的规模以年均25%以上的速度增长，已经成为目前全



国乃至世界高校规模最大的一项大学生课外科技活动。

那么，它究竟有怎样的魅力，可以吸引全国这么多的大学生积极参与？国内知名数学建模专家，清华大学教授，竞赛组委会秘书长姜启源教授在阐述竞赛思路时说：这项竞赛让学生面对一个从未接触过的实际问题，运用数学方法和计算机技术加以分析、解决，他们必须开动脑筋，拓宽思路，充分发挥想象力和创造力，从而培养了学生的创新意识及主动学习、独立研究的能力。从空洞探测、锁具装箱、最优投资组合、灾情的巡视路线、空中交通管理到 DNA 分子排序、血管的分支，这些问题没有现成的答案，没有固定的求解方法，没有指定的参考书，没有规定的数学工具与手段，而且也没有已经成型的数学问题，从建立数学模型开始就要求同学自己进行思考和研究。这就可能让学生亲口尝一尝梨的滋味，亲身去体验一下数学的创造与发现过程，培养他们的创造精神，意识和能力，取得在课堂里和书本上无法代替的宝贵经验。同时，这一切又是以一个小组的形式进行的，对培养同学的团队意识和协作精神必将大有益处。因而也培养了学生的合作精神。通过竞赛，许多取得优异成绩的学生的科研能力明显提高，毕业时受到用人单位欢迎，不少还被免试推荐到国内外高校继续攻读硕士研究生或硕博连读。

中国高等教育学会会长、前教育部副部长周远清在 2010 年高教社杯全国大学生数学建模竞赛颁奖仪式上对这项赛事给予了高度地评价：成功的高等教育改革实践，久盛不衰的一个学科赛事。他指出这个赛事有三个特点：第一是涉及不同学科的一个赛事；第二是培养学生的知识、能力、素质相结合的一个赛事；第三，这个赛事是竞赛跟教学改革密切结合的赛事。全国大学生数学建模竞赛组委会主任李大潜院士在这次颁奖仪式上说道：数学建模不仅是数学走向应用的必经之路，而且是启迪数学心灵的必胜之途。它在应用数学学科中占有特殊重要的地位。同学们通过参加数学建模的实践，亲自参加了将数学应用于实际的尝试，亲自参加了发现和创造的过程，取得了在课堂里和书本上所无法获得的宝贵经验和亲身感受，必然启迪他们的数学心智，促使他们更好地应用数学、品味数学、理解数学和热爱数学，在知识、能力及素质三方面得到迅速的成长。可以毫不夸张地说，数学建模的教育及数学建模竞赛活动是这些年来规模最大也是最成功的一项数学教学实践改革，面向所有专业的大学生，是对教育素质的重要贡献。这个活动得到愈来愈多同学的参与和欢迎，一直方兴未艾，不断向前发展，绝不是偶然的。

“一次参赛，终生受益”是许多参赛同学的共同感受。全国大学生数学建模竞赛是以通信的形式进行，每 3 名大学生组成一队。在 3 天时间内可以自由地收集资料，调查研究，使用计算机、软件和互联网，但不得与队外任何人（包括指导教师在内）以任何方式讨论赛题。竞赛要求每个队完成一篇用数学建模方法解决实际问题的科技论文。竞赛评奖以论文假设的合理性、建模的创造性、结果的正确性以及文字表述的清晰程度为主要标准。

全国大学生数学建模竞赛的竞赛宗旨：创新意识，团队精神，重在参与，公平竞争。

全国大学生数学建模竞赛的指导原则：扩大受益面，保证公平性，推动教学改革，提高竞赛质量，扩大国际交流，促进科学研究。

本竞赛一般在每年 9 月中旬某个周末（周五 8:00 至下周一 8:00，连续 72 小时）举行。

竞赛不分专业，但分本科、专科两组：本科组竞赛所有大学生均可参加，专科组竞赛只有专科生（高职、高专生）可以参加。

同学可向本校教务部门咨询参赛事宜，如有必要也可直接与竞赛全国组委会或各省（市、自治区）赛区组委会联系。

目前除内蒙、青海、西藏、台湾外，全国其他地区均已成立了赛区组委会。

评奖规则：

（1）各赛区组委会聘请专家组成评阅委员会，评选本赛区的一等、二等奖（也可增设三等奖），获奖比例一般不超过三分之一，其余凡完成合格答卷者可获得成功参赛证书。

（2）各赛区组委会按全国组委会规定的数量将本赛区的优秀答卷送全国组委会。全国组委会聘请专家组成全国评阅委员会，按统一标准从各赛区送交的优秀答卷中评选出全国一等奖、二等奖。

竞赛全国组委会联系地址：

北京清华大学数学科学系全国大学生数学建模竞赛组委会（邮编：100084）

电话：+86(10)6278-1785

传真：+86(10)6277-3400

联系人：胡明娅，Email：mhu@math.tsinghua.edu.cn

### 1.2.3 其他数学建模竞赛简介

在美国大学生数学建模竞赛和中国大学生数学建模竞赛举办的如火如荼的同时，数学作为一门基础学科和一种精髓的科学语言，在工程技术，其他学科以及各行各业中所起到的作用也愈来愈受到重视。数学技术已成为高技术的一个极为重要的组成部分和思想库。“高技术本质上是一种数学技术”的观点已为愈来愈多的人们所认同。随之也迎来了应用数学方面竞赛的热潮。

#### 一、MathorCup 全球大学生数学建模挑战赛暨 CAA 世界大学生数学建模竞赛

MathorCup 全球大学生数学建模挑战赛暨 CAA 世界大学生数学建模竞赛（以下简称竞赛）是中国自动化学会主办，中国科学院自动化研究所和校苑数模承办的面向全球大学生的科技竞赛活动，目的在于推动自动化领域中新兴的信息交叉学科的知识普及，培养学生运用数学及其他学科理论和方法解决自动化领域相关问题的能力，同时鼓励广大学生踊跃参加课外科技活动，开拓知识面，培养创造精神及合作意识。

竞赛的前身为校苑数模论坛举办的“MathorCup 全球大学生数学建模挑战赛”，时间通常在每年的4~5月份。竞赛题目共4道，两道题目（A题和B题）一般来源于工程技术、管理科学等方面经过适当简化加工的实际问题，不要求参赛者预先掌握深入的专门知识，另外两道题目（C题和D题）来源于自动化领域中的信息交叉学科（包括图像处理、数据挖掘、人工智能、计算机视觉、模式识别与机器学习等）经过适当简化加工的问题。题目







有较大的灵活性供参赛者发挥其创造能力。

此次竞赛的获奖者不仅仅可以获得一纸证书，还可以拿到中国科学院自动化研究所的暑期夏令营入场券、中国自动化大会免费的参与名额以及中国自动化学会的会员身份。举办比赛的意义不仅仅是给参赛者提供一纸证书，更重要的是为参赛者搭建一个平台，让参赛者通过这个平台学到更多知识，锻炼更多能力，这就是 MathorCup 全球大学生数学建模挑战赛组委会的所思与所想。

网址：<http://www.saikr.com/mathorcup>

## 二、全国大学生电工数学建模竞赛

“中国电机工程学会杯”全国大学生电工数学建模竞赛是由中国电机工程学会电工数学专业委员会与全国大学生电工数学建模竞赛组委会共同发起的，面向全国高等院校学生的一项学科竞赛活动。竞赛意在培养学生运用数学理论和方法解决电气工程领域相关问题的能力，提高学生的创新意识。

电工杯于 2003 年开始举办，在中国电机工程学会的指导下，得到了全国各高等学校的鼎力支持，竞赛举办至今已产生了广泛的影响。2013 年举办的第七届竞赛已有 198 所高校的近 8000 名学生参赛。

该项赛事每两年举行一次，奇数年举行。以队为单位参赛，每队 3 人，专业不限，参赛队伍在指定的 3 天（通常在每年的 11 月末）内完成竞赛。竞赛题目一般源自于电工、近代数学及经济管理等方面实际问题，在经过适当的简化、加工的而来，共有 A、B 两道赛题。参赛者要根据题目要求，完成一篇包括模型的假设、建立和求解、算法的设计和计算机实现、结果的分析和检验、模型的改进等方面的论文。评比标准以假设的合理性、建模的创造性、结果的可行性和文字描述的清晰度为主。竞赛的要求和评比标准与全国大学生数学建模竞赛类似。

网址：<http://dgsx.nedu.edu.cn>

## 三、研究生类数学建模竞赛

### 1. 全国研究生数学建模竞赛

全国研究生数学建模竞赛由教育部学位办与研究生教育发展中心主办，是学位中心主办的“全国研究生创新实践系列活动”主题赛事之一。全国研究生数学建模竞赛是面向全国在读硕士研究生的科技竞赛活动，意在激发研究生群体的创新活力和学习兴趣，提高研究生建立数学模型和运用计算机解决实际问题的综合能力，拓宽知识面，培养创新精神和团队合作意识，促进研究生中优秀人才的脱颖而出、迅速成长，推动研究生教育改革，增进各高校之间以及高校、研究所与企业之间的交流合作。

2004 年举办了第一届全国研究生数学建模竞赛，并于 2006 年被列为教育部研究生教育创新计划项目之一。

以队为单位参赛，每队 3 人，专业不限，时间一般定于 2014 年 9 月中下旬举行，竞

赛题目来源于实际问题,共 A、B、C、D 四道赛题,近年两年增加了 E、F 两题共六道赛题,参赛者任选一题。竞赛规则参照全国大学生数学建模竞赛的规则。

网址: <http://gmcm.seu.edu.cn/>

### 2. 辽宁省研究生数学建模竞赛

辽宁省研究生数学建模竞赛是由辽宁省教育厅和辽宁省人民政府学位委员会主办,东北大学承办。辽宁省首届研究生数学建模竞赛是辽宁省研究生教育创新计划竞赛项目之一,是贯彻落实教育部研究生教育创新计划的重要举措,是面向全省各研究生培养单位在读研究生举办的创新性科技竞赛活动。该赛事旨在激发广大研究生的学习兴趣,提高研究生建立数学模型和运用计算机解决实际问题的综合能力,提高研究生创新的质量。

该项赛事从 2013 年开始举办,辽宁省各研究生培养单位在读研究生均可参加。研究生以队为单位参赛,每队 3 人,专业不限。参赛各队不要求指导老师,由研究生自主参加,旨在突出研究生自主创新。竞赛内容一般来源于工程技术和科学管理方面经过适当简化加工的实际问题,共 A、B 两道赛题。竞赛时间一般是三天(一般在每年的 5 月中旬举行)。

竞赛不要求参赛者预先掌握深入的专门知识,适合我国多数学科研究生的水平,使参赛队伍在规定时间内有充分发挥聪明才智和创新精神的余地,而且一般要先建立数学模型并用计算机求解,但不要求在此期间内能完全解决问题。参赛者应根据题目要求,完成一篇包括模型的假设、建立和求解、计算方法的设计和计算机实现、结果的分析和检验、模型的改进等方面的论文。竞赛评奖以假设的合理性、建模的创造性、结果的正确性和文字表述的清晰程度为主要标准,并特别重视创新性和实用性。

网址: <http://www.neu.edu.cn/>

### 四、美国高中数学建模竞赛

美国高中数学建模竞赛(High School Mathematical Contest in Modeling)简称 HiMCM。始于 1999 年,由美国工业与应用数学学会(SIAM)和美国运筹学会(ORSA)发起,美国数学及其应用数学联合会(COMAP)主办的一项面向高中生的科技竞赛活动。竞赛得到了美国国家科学基金会(NSF)、运筹和管理科学研究所(INFORMS)、美国数学协会(MAA)和美国全国数学教师委员会(NCTM)的资助。

这项竞赛借鉴了美国大学生数学建模竞赛的模式,结合中学生的特点进行设计的。竞赛队最多由 4 名高中生组成,配备一位指导老师,在指定的 17 天(通常由 11 月第一周的星期五开始)内,由参赛队自己选定的连续 36 个小时完成竞赛。赛题分 A、B 两题,均源于实际问题,赛题为来自现实世界的 A、B 两个实际问题,参赛队任选一题,最终以论文的形式上交。竞赛其他要求和论文评比标准与美国大学生数学建模竞赛相同。

网址: <http://www.comap.com>





## 五、各地区数学建模竞赛

### 1. 华东杯数学建模竞赛

华东杯数学建模竞赛是一项由复旦大学数学科学学院发起，华东地区数学建模联盟组织开展的竞赛。竞赛意在激励学生学习数学的积极性，开拓知识面，提高学生独立分析问题、建立数学模型、运用计算机技术模拟解决实际问题、论文写作等综合能力，鼓励广大青年学生在基础及应用学科研究中推陈出新，促进数学教育改革，培养学生的创造精神及合作意识，塑造同学们的科创意识与团队精神，为同学们将来能更好地走上社会、服务社会打下坚实的基础。

竞赛从 1999 年开始举办，在全国高校中享有较高声誉。竞赛题目一般来源于工程技术和 管理科学等学科领域经过适当简化加工的实际问题，共 A、B、C 三道赛题。参赛对象必须为在校大学生，选手以队为单位参赛，每队不超过 3 人，专业不限，参赛者从中任选一题完成论文。竞赛规则参照全国大学生数学建模竞赛的规则。

网址：<http://www.mmmm.fudan.edu.cn/>

### 2. 苏北数学建模联赛

苏北数学建模联赛是由江苏省工业与应用数学学会、中国矿业大学、徐州市工业与应用数学学会联合主办，中国矿业大学理学院协办及数学建模协会筹办的面向苏北及全国其他地区的跨校、跨地区性数学建模竞赛，目的在于更好地促进数学建模事业的发展，扩大中国矿业大学在数学建模方面的影响力；同时，给全国广大数学建模爱好者提供锻炼的平台和更多的参赛机会，鼓励广大学生踊跃参加课外科技活动，开拓知识面，培养创造精神及合作意识。

2003 年 3 月份，中国矿业大学数学建模协会便开始组织筹划苏北地区首届数学建模联赛，因非典未能顺利举行。自 2004 年 5 月 1 日—5 月 4 日成功举办“首届苏北数学建模联赛”以来，参赛规模在不断扩大，现已有 14 个省市地区 40 多所高校的 2700 多名学生参加。

参赛队由三名具有正式学籍的在校大学生（本科或专科）组成，参赛队从 A、B、C 题中任选一题完成论文，本科组和专科组分开评阅。竞赛按照全国大学生数学建模竞赛的程序进行，网上报名，报名时间为每年 4 月 1 日—4 月 29 日，竞赛时间为 5 月 1 日—5 月 4 日。苏北数学建模联赛组委会聘请专家组成评阅委员会，评选一等奖占报名人数 5%、二等奖 15%、三等奖 25%，如果有突出的论文将评为竞赛特等奖，凡成功提交论文的参赛队均获成功参赛奖。

网址：<http://slxy.usts.edu.cn/mm/www.cumcm.html>

### 3. 东北三省数学建模联赛

东北三省数学建模联赛是由黑龙江、吉林、辽宁三省高校联合发起的面向大学生、研



究生的赛事。发起这一赛事的目的是进一步普及数学建模教育，培养学生应用数学知识解决实际问题的能力，激发学生学习数学的积极性。

联赛从2006年开始举办，竞赛时间大约在每年4月25日8:00至5月8日15:00，竞赛题目一般来源于实际问题，共A、B、C、D四道赛题。参赛队伍由三名具有正式学籍的在校大学生（本科或专科）或研究生组成，参赛者从中任选一题完成论文，本科组、专科组和研究生组分开评阅。竞赛规则参照全国大学生数学建模竞赛的规则。

网址：<http://www.mcm.edu.cn>

### 4. 华中地区大学生数学建模竞赛

华中数学建模联盟赛是在湖北省竞赛组委会，武汉工业与应用数学学会，由华中地区数学建模联盟会发起并组织开展的竞赛，以“提高华中地区数学建模能力、发展并壮大数模事业”为宗旨。竞赛目的在于，提高学生独立分析问题、建立数学模型、运用计算机技术模拟解决实际问题、论文写作等的综合能力，提高各高校大学生数学建模水平，加强各高校数学建模能力。

竞赛从2008年开始举办，竞赛时间大约在每年5月8日上午8:00至5月12日上午8:00，竞赛题目一般来源于实际问题，共A、B两道赛题。参赛对象主要针对华中地区高校的在校大学生，同时也欢迎非华中地区高校在校大学生报名参加。参赛队伍由2~3名具有正式学籍的在校大学生组成，参赛者从中任选一题完成论文。竞赛规则参照全国大学生数学建模竞赛的规则。

网址：<http://www.shumo.cn/>

## 六、各学会组织的数学建模竞赛

### 1. APMCM 亚太地区大学生数学建模竞赛

APMCM 亚太地区大学生数学建模竞赛是由校苑数模网主办的科技竞赛，旨在进一步普及数学建模知识，强化学生应用数学解决社会、自然的相关问题，并增强计算机的理论和编程能力。为亚太地区学生提供良好的数学建模家园，并为学生创造更多参加数学建模竞赛的机会。

2010年举行了首届APMCM 亚太地区大学生数学建模竞赛，竞赛时间大约在每年的11月或12月，先后分中国赛区与亚太区两个阶段，只有中国赛区一等奖的队伍可以晋级亚太区决赛的评审阶段。参赛对象为亚太地区高校全日制在校专科生、本科生、研究生，高校毕业生（社会工作者）、数学建模爱好者。以队为单位参赛，每队3名学生。竞赛题目共A、B、C三道以中文和英文的格式给出，参赛者任选一题，提交一份英文论文。竞赛规则参照美国大学生数学建模竞赛的规则。

网址：<http://apmcm.mathor.com/>





## 2. “认证杯”数学中国数学建模网络挑战赛 (TZMCM)

“认证杯”数学中国数学建模国际赛由内蒙古自治区数学学会和全球数学建模能力认证中心共同主办，数学中国和第五维信息技术有限公司协办的全国性数学建模活动。数学中国成功获得全球数学建模能力认证中心的授权，其目的是激励学生培养数学建模的能力，明确数学建模能力要求及范围，为数模社会效益化积累人才。

从 2008 年起开始举办，赛题一般来自工程技术和科学管理方面经过适当简化加工的实际问题，共 A、B、C、D 四道赛题，其中 D 仅限中学组\专科组选做，本科组不可选做。竞赛分为“建模基础”和“模型改进”，时间分别在每年的 4 月中旬和 5 月中旬，其他要求与全国大学生数学建模竞赛相同。

网址：<http://www.tzmcm.cn/index.html>

## 3. “认证杯”数学中国数学建模国际赛

“认证杯”数学中国数学建模国际赛与 TZMCM 都是由数学中国协办的，目的都是为了激励学生培养数学建模的能力，明确数学建模能力要求及范围，并为数模社会效益化积累人才。区别在于小美赛是为参加美国大学生数学建模竞赛做准备。

网址：<http://mcm.tzmcm.cn/>

## 1.3 数学建模活动与能力培养

一滴甘露能滋润一个生命，一抹阳光能温暖人的心房，一段不凡的经历可铸就人的辉煌。

在数学建模这个平台上，“建模人”用心书写着自己的人生之路。流过的汗水代表拼搏，喜悦的泪水象征成功；激烈的辩论是探求知识，真诚的合作滋润你我……

无数次的比赛带给他们无数次惊喜，而无数次喜悦的背后又有多少局外人难以想象的艰辛；无数次的比赛磨砺了他们的意志，也丰富了他们的人生经历。在这个平台上，他们学会了学习，这种学习是融知识与应用于一体的学习；他们学会了思考，这种思考是集理论与实践于一体的思考；他们学会了合作，这种合作是他们成功的助推器，在合作中促进了成功，也在合作中产生了友谊；他们拥有了一种心态，这种心态是他们成功的基石，因为他们胜而不骄，败而不馁；他们没有“力拔山兮气盖世”的豪言壮语，却有着“路漫漫其修远兮，吾将上下而求索”的精神；他们没有惊天动地的经历，却有着在遇到困难时将“建模”进行到底的勇气。

本节我们主要从以下三个方面讲述数学建模活动与能力培养的关系：数学建模对学生就业、升学以及出国的帮助；企业对待数学建模活动的态度；数学建模活动对参赛者将来从事科研和工作的影响。

### 1.3.1 数学建模与就业、升学、出国

学习数学建模能够接触到各种国内外的数学软件,如 MatLab、SAS、Lingo 等;能够拓宽解决问题的思路与方法;能够提高解决实际问题的创新能力和动手能力以及科研能力;能够体验撰写论文流程;能够锻炼学生的抗压耐压能力。因此,学习了数学建模特别是参加了国内外的数学建模大赛,至少让别人明白你已基本具备了上述能力。

一个人基本具备了这些能力,就有了实际工作的能力。在就业市场上,很多单位特别需要有思想能动手还能吃苦耐劳的工作者。在中国当前答谢教学教育中真正具备这些能力的学生并不为多数。建模学生正是以他们的博学多才、敢想敢干、坚强意志而深受用人单位青睐。以华中农业大学为例,自 2005 年以后该校数学建模成绩突飞猛进。只要参加过数学建模的学生都找到了很好的工作。阿里巴巴、百度、搜狐、华为、腾讯、京东等企业均有他们的学生。

学习数学建模,不仅能够学习到很多建模、处理数据的方法,更能培养一个学生思考问题、解决问题的良好习惯。无论是自然科学还是社会科学甚至是人文科学等,抽取问题背景和解决的问题,剩下的都会或多或少的归结为数学问题。数学是任何学科科学研究的基础和工具。一个学生一旦掌握这种研究问题的方法和意识,那么他在科学研究中就会容易取得成就。作为高校和研究生导师,非常愿意录取到这样的学生。华中农业大学每年参加国赛学生有近一半的学生进入到北京大学、清华大学、北京师范大学、中国人民大学等“985”高校进行深造。有近三分之一的学生本科毕业后就进入到美国、英国、德国、澳大利亚、新加坡等国的著名高校进行深造,大部分学生取得了资助。在华中农业大学,参加数学建模的主要动力就是升学和出国。很多出国培训机构都和该校数学建模基地进行了合作。该校数学建模团队正努力将数学建模基地打造为出国基地。

### 1.3.2 企业中的数学建模问题

本世纪以来,中国经济高速发展。中国与世界先进发达国家的差距越来越小。经济市场化、全球化的步伐越来越快。数据信息的海量化和复杂化程度越来越高。任何企业在高速变化的全球经济中面临的竞争和挑战将会很大,但同时机会也很多。如何科学的进行决策,以期获取最大的利润是企业的生存之本。作为一个企业,需要在市场竞争分析、消费需求分析、生产优化控制、运输储存、产品开发、资源管理以及人员调配等诸多环节都要进行系统优化。而这些优化是离不开数学建模的。随着大数据时代的到来,企业更需要大数据处理的能力,而大数据的处理需要有硬件和软件,而软件就是数学建模和相应配套的计算方法。

数学建模正是教给学生如何运用数学建立大数据处理中、企业生产决策中所需要的数学模型,并编写相应的计算方法。具备这些能力的学生无疑是企业所青睐的。数学建模人才的培养与社会需求紧密相连,因而具有旺盛的生命力。





企业中的数学建模问题，大致可分为五大问题。一是预测预报问题，包括产品销售、交易期望、生产前景等。二是评价与决策，包括实施方案的风险评估、项目的选择、绩效的评价等。三是分类与判别，包括消费群体的分类、产品归属的判别等。四是关联与因果分析，包括产品质量控制、市场营销等。五是优化与控制，包括生产流程控制、产品定价问题、工程预算问题、规划设计问题等。


数学建模的问题本身就来自于现实世界，来自于企业。学习好数学建模无疑能够在企业的生产活动中发挥重要的作用。掌握了这种技术的人也无疑将掌握企业的命脉。

### 1.3.3 数学建模对科研和工作的影响

数学在生活当中无处不在，数学的能力考查并不仅仅是单纯数学知识层面深浅的考查，更多的是数学思维能力，数学的应用能力的考查。数学建模的实践，就是引导学员们发现实际生活当中的数学规律，学会应用数学方法去解决生活当中的问题，从而使自己的思维能力得到很好地锻炼。

所以说，数学建模学习到的其实是一种技能，一种可以伴随人一生的思维能力。包括逻辑思维能力，逆向思维能力，创新能力，快速自学能力，文字表达能力，以及团队协作能力，等等。

数学建模的学习，培养了学生“学数学，用数学”的意识和能力，包括查阅资料的能力、文献综述的能力、模型建立的能力、问题分析的能力、计算编程的能力、科研写作的能力以及超强的自学能力。

有了这些能力，学生就有了创新能力和动手能力，就有了较高的科研潜能和素质，也同时具备了较强的工作能力。科研工作需要一个具有肯吃苦、善思考、勤动手、能反思等素质的人来担当。数学建模的培养就是为了培养这些素质，因而学习数学建模就是为了培养工作能手和科学骨干。 

## 第2章 数学建模中的团队合作

本章主要介绍比赛中有关团队合作的一些要点和比赛中的一些技巧，主要内容有团队的组建和分工；比赛前和比赛中的时间安排；文献管理器 Zotero 的使用；论文各部分的撰写要点，以及论文排版工具 LaTeX。

### 2.1 组建团队

数学建模比赛不光涉及数学，同时还涉及到编程、写作以及其他学科的各种知识。参加数学建模比赛，最重要的是队伍的团结和配合。在比赛中，团队中的三人会有分工，各成员负责的任务侧重点不一样。一种最常见的分工方式是：写手，程序员和第三人。写手主要负责论文的大部分写作，程序员则主要负责程序的编写和其他数值工作，第三人则处理其他的一些琐碎杂事，并协助其他两队员完成他们的工作。然而分工有时也并不太可能非常明确，比如在比赛中，写手也可能参与程序的编写，而程序员也可能会负责部分论文的写作。但每个部分都要有一个主要的人负责，大家要最终听那个负责人的，省去不必要的争论。最好的队伍不是三个人都是建模高手，而是三个人都能 cover 一个方面，互相又在那个方面比较服气对方。不过团队的所有队员都应该共同参与问题背景的研究，模型的建立，摘要的写作及论文的最终修改。

在参加竞赛前每一名队员应考虑自己在团队中扮演什么样的角色，承担什么责任。想一想你能担当什么样的角色。本节中将分别阐述建模比赛中分工的三种角色需要注意的问题。

#### 2.1.1 程序员

数学建模竞赛特别强调计算机编程解决实际问题的能力，尤其最近几年。比如 2014 年 MCM 的 A 题“除非超车否则靠右行驶的交通规则”，几乎所有特等奖论文都是运用计算机编程模拟出不同的交通规则。复杂的数学模型需要计算机来实现计算或预测，这些程序可能长至数千行，也可能短至数行。因此参赛队中必须要有至少一名程序员，程序员需要非常熟悉计算机编程。要能够选择适当的算法将数学模型转化为计算机语言，根据不同的方案或参数计算出结果，并比较各方案或参数。对于程序员来讲，主要需要用程序实现以下工作。

- 数据的可视化：将计算结果以曲线、曲面、直方图等形式展现，使其更直观的呈现问题、模型和结论。如果程序员主要使用 C/C++，Java 等不方便出图的语言，







那么程序员可能还得借助于其他计算机语言或软件, MatLab, Python, Mathematic, Gnuplot, MicrosoftExcel 等都可以很好的实现数据的可视化。

- 数值算法: 程序员需要掌握一些常用的数值算法, 诸如插值、拟合、求线性方程组、求解微分方程数值解。幸运的是, MatLab 等语言包括各种各样的内置函数, 能够非常方便地实现这些数值算法。比如可以直接调用 `interp1` 函数来实现插值。
- 高级算法: 程序员还需要掌握一些数学建模中常用的模型和算法的程序实现方法, 如动态规划等优化算法, 图论算法, 蒙特卡罗算法, 图像处理算法, 元胞自动机方法等。这些算法赛前不准备是没有办法在比赛中很好运用的, 因此在赛前每个常用的算法都最好自己去编程实现一下。

程序员最好选择一种或两种自己最最熟悉的语言, 并且所选择语言能够完成以上工作, MatLab, Python, Mathematic 都是不错的选择, 这些语言既能够实现算法, 又能够可视化数据。虽然比赛中并不限制使用的计算机语言种类和种数, 但笔者还是建议同一个参赛队最好能统一使用一种或两种语言, 以使代码具有较好的重用性。在程序的编写过程中, 应注意

- 模块化: 将长的代码尽量拆成短的函数, 以方便程序的快速调试通过。比赛时, 宝贵的时间用来长时间调试是不值得的。
- 参数化: 在主程序的头部定义好每个参数, 以方便测试不同的方案或参数 (只要更换参数值即可)。
- 易读性: 程序中的变量名最好有具体意义, 比如表示距离的变量名可以定义为 `dist` (`distance` 的缩写)。程序关键部分也要有恰当的注释。

笔者提醒各位, 编程方面的能力和习惯不是一朝一夕可以练就的, 需要长期刻苦的训练, 一个人只需要掌握一门语言就行了, 但不必要做到精通。

## 2.1.2 写手

虽然程序员负责的工作是至关重要的, 但是最终写出程序的东西并不意味着会起作用。各参赛队最终提交的作品也就只有论文, 不在论文中表述, 就不会对比赛结果产生任何作用。论文是评委能够唯一看到的成果, 也是评委评判的唯一根据。所以写手的水平直接决定了获奖的高低, 重要性也不言而喻了。论文的写作可以说是数学建模竞赛中最重要的工作, 写作甚至比解决问题本身更为重要。正如一位特等奖作者给我的邮件中写道 “A well written paper is more important than groundbreaking results.” (笔者通俗地译为: 结果好, 不如论文写得好)。

在数学建模竞赛有限的时间里 (美赛 4 天, 国赛 3 天), 自始至终都应伴随着写作, 论文的正式写作应尽早开始。许多参赛队往往低估了论文写作所需时间, 几乎大半时间花在了模型建立和模型求解上, 最后论文的写作却是草草了事。这样自然是写不出条理清晰的好论文。论文的初稿越早完成越好, 论文的完整初稿应该在比赛结束前一天完成 (国赛则至少提前一个晚上完成论文的初稿), 留一天的时间用于论文的修改润色和摘要的写作。



论文必须要写得非常非常清楚，要呈现的内容要尽量简练而准确的表达出来，特别是在美赛中。一篇优秀的论文其语言都是言简意赅的。完全没有必要让模型看起来更复杂，水平高的写手可以使复杂的模型看起来更简单清楚。作为写手，其目标就是使一切尽可能的简单，逻辑清晰。写手时刻要从评委，也就是论文阅读者的角度考虑问题，在全文中形成一个完整地逻辑框架。要善于适当地利用图表公式把模型和结果以最清晰的方式展现在论文中。

虽然论文的写作由专人负责，但论文的最终修改和完成应该是所有队员都要参与的。团队中的每个人都需要来评评论文的每一个部分，在做评判的时候不掺杂个人色彩。完全没有个人色彩的写作是很困难的，但竞赛却要求这样。作为写手，你需要写一大堆文字，然后找队友们评判，并得到反馈，再修改，反复这样。写作，重写，反复重写。

作为准备，参赛队员，特别是写手最好阅读近些年竞赛的特等奖论文，这是非常重要的。这是唯一的一个渠道来感知什么样的论文才是好论文。要想写出好的数学建模论文，首先得知道什么样的论文在评委们的眼中是好论文。当你阅读完一篇特等奖论文后，尝试猜猜评委们的观点，总结这篇论文为何得到了评委认可的理由。

对于写手，还需要掌握一种排版工具，笔者推荐使用 LaTeX 作为写作的工具。关于 LaTeX 的使用，本章将在第 5 节中简单介绍。当然，对于没有使用过 LaTeX 或者不太熟悉 LaTeX 的写手，笔者并不建议把时间浪费在调试 LaTeX 上。Microsoft Word, Open office 等也都是不错的选择。

### 2.1.3 第三人

写作是在竞赛中取得好成绩的基石，编程也同样非常重要，但是还有许多其他重要的工作。为了让团队获得较好的成绩，所有队员都必须全力以赴，因此，第三人必须寻找一些事情去做。

第一个大的工作就是资料检索。找到尽可能多的相关问题的资料，尽可能多的解决问题的方法。为了能够在竞赛中应用，资料检索通常是非常具体的。从数学书籍中挖掘出一些能够在计算机上应用的东西。到所知的数值分析算法库，寻找能够用到的东西。第三人还应该参与写作。作为写手的一个辅助，确保每一件事都非常清楚，并帮助写手完成论文的写作，比如文献的整理。第三人不仅是论文的读者、评论者，而且应该是论文的另一主要合作完成者，每个人都应该参与论文写作，包括程序员。

如果可能，第三人也要能够编写程序。如果一个队中有两个人具有编程能力，那是一个非常不错的组合。

### 2.1.4 团队合作

在数学建模竞赛中，一个成功的团队必须使得每个人在每一分钟都能发挥作用。这一点尤其要提醒团队中的低年级成员。如果你是一个大一或大二的学生，并且团队中有一名





或者两名高年级的成员，当其高年级成员表现得像专家一样的时候，那么寻找一条为团队做贡献的途径并不容易。你必须时刻保持自己参与，因为在短短的三四天中有太多的工作需要做了，如果没有你，或者你仅仅开了一半的油门，你的团队都不太可能很好地完成比赛。如果你发现自己正处于这种状态下，你应该找到你的队友们，并直截了当地对他们说：“我应该做些什么？我又如何能为团队作出贡献？”。

如果你是一个团队中的高年级成员，不要事事都亲自去做。与你的队友一起工作，不要担心把工作授权给别人。如果你是一个大一或大二的学生，你应该做什么？你如何能为团队做出贡献？在团队中，你应该扮演一个怎样的角色？首先你要保证你可以理解你的团队建立模型的每一个细节处的数学知识。

假设你的队友们想出了一些高级的算法来解决问题：要在论文中表述他们的想法，可以尝试首先清楚地表述给你。让他们解释每一件事，直到他们能真正说得清清楚楚。别忘了，你有可能以新手的角度来看待问题，所以不管你是不是论文撰写者，都要为论文把关。寻找一些不合理的，逻辑混乱的，表述含糊，不清楚的地方，然后做出一些质疑，直到被正确改正。低年级的同学往往更适合做一个团队的置疑者。问所有你能想到的问题。为什么要做这一步？这是什么意思？这个方程从何而来？你们最终的论文，应该和一本教科书的风格一致，应该把你们使用的方法教给读者。当然评估一本教科书的最好办法，就是拿给一个班的新同学来看，看看他们是否能完全理解。不要让你的队友敷衍你：如果他们不能给你清楚的解释，他们也同样不能期望在论文中表达清楚。

### 2.1.5 赛前模拟

赛前模拟对于在比赛时顺利地完论文是非常重要的，特别对于初次参加比赛的同学。赛前模拟绝不是为了押题。赛前模拟，一方面能提前发现比赛中可能会出现的问题，并想好应对的方案，以避免在比赛时再浪费时间来应对；另一方面能够磨合团队的合作，发现各自的优缺点，以便在真实的比赛中实现更好的分工和合作。

参赛之前最迟一个月就应把队伍组好了，以便有充足的时间进行赛前模拟。这一个月的时间就是三个人一起做往年的赛题。先不要看相应的特等奖论文，要自己动手，把整个流程都走一遍，最好把文章都完全写出来。待整个流程都走完后，再看特等奖论文，找找自己的论文和特等奖论文的差距。这一个月一定要保持一个良好的竞技状态，到真正参赛的时候才能拿出最好地水平。新加坡国立大学的王皓（2007 年 MCM 特等奖获得者）在给我的邮件中就曾回复说：“我们当年参赛之前一个月，每周做一套之前的题（那时候我们在新加坡国大，一月份开学，一直是在上课时间，不是寒假，所以每天只能抽出晚上的时间来做模拟），做了三套，手感就来了。比赛在星期五，我们头一个星期日做完最后一套模拟题之后，就给自己放了四天的假，养精蓄锐，最后参赛的时候整体的状态非常好。”从一个特等奖队的成长中可以看出，赛前模拟是非常有必要的，而且能够使团队的水平迅速提高。



## 2.2 时间安排

数学建模比赛时间有限，一个合理的时间安排和分配是非常重要的。这里我们以四天的美国大学生数学建模竞赛为例，给出一种比较合理的时间安排。

### 2.2.1 赛前准备

在比赛前，各参赛队应该准备好一切比赛可能会用到的东西，而不是在比赛过程中再去准备，不仅浪费了宝贵的比赛时间，也搞坏了心情。参赛队应该准备好以下几方面：

- 食物：赛前要准备好比赛中吃的东西，可以多买一些自己喜欢吃的食物和饮料。由于比赛可能需要熬夜，因此干粮，水果等也是必不可少的。
- 计算机：提前清理一下计算机，保证计算机以最佳状态运行。安装并设置好各种数学软件。通过代理等手段实现翻墙，保证计算机能够顺利访问 Google 等网站。
- 书籍和论文：手头最好也备一些常用的书籍，除高数，线代，概率论等课本外，可以备一些编程，模型，算法方面的书。虽然备了不一定有用，但是放在那里总归是心里踏实。对于平时读过的特等奖论文，最好也打出几篇来，在写作时可以参考参考，特别是写摘要的时候。
- 数据库：准备好查找文献的期刊网入口，无论是中文的知网，维普，还是英文的 SCI, Springer 等都要提前找到，总之最后不要影响到比赛中查找文献。

### 2.2.2 第一天：开始比赛

在比赛的第一天，拿到赛题后面临的第一个问题就是选题。参赛队需要在较短的时间内从 A, B, C 三题中选择一题进行解答。A, B 题属于 MCM, C 题属于 ICM。在读完全部赛题并对问题有一定的背景认识后，参赛队应该根据队员各种技能的优势和兴趣选择合适的赛题。如果队员在选题上的意见一致，那么就可以开始进入做题阶段。如果选题意见不一，那么可以花两三个小时对各问题的背景做进一步的了解（通过文献检索和阅读），再行定夺。

首要任务是仔细读题，并进行头脑风暴，列出能想到的所有可能的解题方案。仔细读完赛题后不要急于讨论，每个队员对问题都必须有一个独立的思考，有一些原创的想法更好。每个队员都对问题经过独立思考后，再进行小组讨论。然后开始问题的正式研究，检索与当前问题相关的文献。阅读相关的文献将有助于你对问题的正确认识，并以正确的方式思考问题。第一天的整个晚上，你们需要认真努力的研究问题，建立模型，编写程序。

### 2.2.3 第二天：建立模型

第二天凌晨和上午稍作休息后，继续工作。第二天可能是四天中最关键的一天，这一





天的时间绝大部分将花在问题的研究，模型的建立和程序设计上。并开始写作。应该在第二天完成问题的大部分解答，并尽可能早的开始写作。尽可能在当晚从模型中获得一些初步结果，一旦有了初步结果，就可以睡觉了。

## 2.2.4 第三天：写作和修改

第三天应该花在结果的求解和论文的主体写作上，基于第二天晚上得到的初步结果，花一天时间来改进方法并重新计算结果。在第三天，团队应该共同写作完成论文的主体部分。写手可以为其他成员指派任务以更快的完成论文的写作。直到大部分写作完成，所有程序编写完成后才能睡觉。

## 2.2.5 第四天：写作和润色

第四天的早晨应该已经完成了所有程序（没有新程序要编写了）。队员继续完善论文的其他部分，如论文的优缺点，结论等。在你写论文的缺点时，你可以尝试一些改进方案来避免这些缺点。第四天的下午就应该完成论文的初稿了，可以打印出来并仔细修改。到晚上，所有结果和论文都应该完成，并开始着手写摘要。

第四天晚上到第五天早晨结束比赛前，这段时间大部分用来写摘要，即使你还有一些工作没有完成，这时也得停下来。如果时间剩余，再润色一下论文，完善一下附录。当论文和摘要全部完成后，再给论文定一个标题。

## 2.3 文献管理器

在数学建模比赛中，通常会涉及大量的文献管理和引用。为了方便文献的管理和引用，常常会使用文献管理工具来管理文献。常用的文献管理器有 Endnote 和 Zotero 等，这里我们介绍 Zotero 来管理文献。Zotero 是开源的文献管理工具，可以方便的收集、组织、引用和共享文献。由安德鲁·W·梅隆基金会，斯隆基金会以及美国博物馆和图书馆服务协会资助开发。

### 2.3.1 Zotero的安装和配置

从 <http://zotero.org> 下载 zotero 安装文件及浏览器插件，浏览器插件支持 Chrome、irefox 和 Safari。

如果你使用 Word 排版，那么你还需要从网上（网址：[https://www.zotero.org/support/word\\_processor\\_integration](https://www.zotero.org/support/word_processor_integration)）下载并安装 Word 支持包。这将在 Word 的加载项中增加一个如图 2-1 所示的图标。



图 2-1

在 <http://zotero.org> 注册账号,并在 Zotero 软件的工具-首选项-同步中登入账号,登入之后,保存在 Zotero 文献库中的文献条目和笔记目录将同步到网络。

### 2.3.2 文献条目的保存

Zotero 的最大优点是能够对在线文献数据库网页中的文献题录直接抓取。

如我们打开数据库中的某篇文献的网页,在安装有 Zotero 的 Firefox 或 chrome 的地址栏末端会出现一个论文或书籍标志。点击这个标志,即可将当前页面中的文献条目保存到 Zotero 中。

如果你访问的页面包含多篇文献,比如 Google 学术论文,在安装有 Zotero 的 Firefox 或 chrome 的地址栏末端会出现一个文件夹标志,点击可选择一篇或多篇文献条目保存。如图 2-2 所示。



图 2-2



### 2.3.3 文献分类

Zotero 中的所有文献条目都保存在我的文献库中,这些文献条目可以划分到不同的文献分类中,使文献条目的保存更具条理性。通过点击如图 2-3 所示的图标可以增加一个文献分类。



图 2-3

### 2.3.4 文献引用

如果你使用的是 MSWord 来作为论文排版工具,你会发现在 Word 的加载项中有一些 Zotero 的相关图标。

如果你想在 Word 文档中引用一篇 Zotero 文献库中的文献,可点击右图 2-4 所示的图标。如果这是该文档第一次插入文献,你需要选择文献的格式。



图 2-4

在论文的末尾,我们需要生成引用过的文献列表,这里可点击如图 2-5 所示图标。



图 2-5

如果你使用 LaTeX 来作为论文排版工具,你可以从 Zotero 中直接导出所需要的 BibLaTeX 文件,操作方法是:选中一篇或多篇文献条目,右击后选择导出条目,选择导出



格式为 BibLaTeX 即可导出 bib 格式的文件。

## 2.4 撰写论文

### 2.4.1 标题 (Title)

论文的标题是给评委的第一印象, 是以最恰当, 最简明的词语反映论文中最重要的特定内容的逻辑组合。对于往年的数学建模特等奖论文, 标题的确定通常是以下几种方式:

- 以所使用的主要方法或模型来构造论文标题: 如 2014 年 MCM 的 A 题, 一篇特等奖论文的题目为 “Freeway Traffic Model Based on Cellular-Automata and Monte-Carlo Method”; 再如 2012 年 ICM, 一篇特等奖论文的题目为 “Crime Ring Analysis with Electric Networks”。
- 以结论来构造论文标题: 如 2014 年 MCM 的 A 题, 一篇特等奖论文的题目为 “Keep Right to Keep Right”; 再如 2004 年 MCM 的 A 题, 一篇特等奖论文的题目为 “Not such a small whorl after all”。
- 直接以赛题的题目或要解决的问题作为论文的标题: 如 2014 年 MCM 的 A 题, 一篇特等奖论文的题目为 “he Keep Right Except To Pass Rule”; 再如 2003 年 MCM 的 B 题中, 一篇特等奖论文的题目为 “The Gamma Knife Problem”。
- 以幽默的方式构造标题, 吸引评委: 如 2003 年 MCM 的 A 题, 一篇特等奖论文的题目为 “You Too Can Be James Bond”; 再如 2007 年 MCM 的 A 题, 一篇特等奖论文的题目为 “When Topologists Are Politicians”。

以上四种论文标题的确定方法的界线有时也并不非常明确。比如 “When Topologists Are Politicians”, 不仅以幽默的方式构造标题, 同时也暗含了论文中所使用的主要方法为拓扑学。读者可以根据实际情况选择一种标题的确定方法或几种标题的确定方法的组合来确定数学建模论文的标题。

### 2.4.2 摘要 (Summary)

摘要无疑是论文中最重要的部分。摘要应该最后书写。再重申一遍: 在论文的其他部分还没有完成之前, 你不应该书写摘要。一个理想的时间安排是把交卷前一天的时间拿出来书写摘要。

摘要应该使用简练的语言叙述论文的核心观点和主要思想。如果你有一些创新的地方, 一定要在摘要中说明。进一步, 你必须把一些数值的结果放在摘要里面, 例如: “我们的最终算法执行效率较一个简单的贪婪算法提高 67.5%, 较随机选择算法提高 123.3%”。

理想的摘要长度是很难确定的。你必须把所有的核心观点包含在摘要里面, 但是简洁是非常重要的。一般情况下半页左右比较合适, 绝对不要超过 2/3 页。

摘要 (甚至是整篇文章), 应该由整个团队合作完成。一种实现方式是, 每个队员单

独地花一个小时（至少）时间写一个自认为最好的摘要。然后，大家聚到一起，相互阅读这些摘要。再经一翻讨论后，团队在此基础上，共同写一个新的摘要。

### 2.4.3 引言 (Introduction)

在引言中，你可以按照你自己对问题的理解重述问题。从一个建模问题中，几乎每一个参赛队都可以找到不同的“模型”来进行解决。赛后当你阅读其他参赛队的论文的时候，你会惊讶地发现你们解决问题的方法非常不一样，甚至，有的时候你会发现你们解决的问题也是截然不同的！

因此，你在引言中要将你对问题的理解以及你的工作所要解决的问题表述清楚。为了确保所有成员都赞同对问题的理解以及要做的工作，可以写出引言并由所有队员各自阅读直到意见达成一致。在引言中也可以阐述一些问题的背景，或者展示一些你在研究问题过程中学到的东西。需要注意的是，评判你们论文的是数学教授，如果你们不展示出你们理解如何用教科书上的方法来解决这个问题，他们可能会不爽。无论你最终是选择教科书上的方法还是某些更有创造性的方法，在引言中都需要提及传统教科书上的方法，这样这些教授便会知道你是个做作业的好孩子。

引言通常应该在比赛第一天下午首先书写。它可以帮助确保团队所有成员的工作同步。

### 2.4.4 模型 (The Model)

数学模型的目的在于对真实世界的预测，并帮助你更好的理解真实的世界。论文主体的第一个部分通常用来描述模型。绝大多数问题，都可细分为三个部分：模型，解决方案和验证方法。基于一定的目标，你们需要建立一种模型或方法来实现这些目标。这些目标可以是找出最佳的交通规则，也可以是预测海平面的上升。数学模型的目的在于预测出不同方案将引发的结果。比如对于最佳交通规则问题，需要预测不同交通规则给出的流量和安全性，比较后最终评定出最佳交通规则。

优秀的论文一般会包含一系列的模型，从最简单的模型到较复杂的模型和更真实的模型。对于任何一个问题，应该总是首先尝试建立一个非常简单的模型，简单到能够用笔和纸就能求解。一般来说，复杂的模型都呈现于计算机中，所以我们面临的挑战是将程序代码翻译成文字，使得每一步都能自圆其说。对于一些连续问题的建模，建议要对如何求解微分方程有一个清楚的理解。比如对于对动物种群的问题，我们期望能够写出微分方程来正确的描述捕食者和猎物种群之间的关系，然后进行数值积分求解微分方程。一般来说，对于离散问题，你需要熟悉如何产生具有不同性质的随机数集合，这对于构建用于检测你的算法及测试数据集很有帮助。

### 2.4.5 解决方案 (The Solutions)

解决方案是论文的第二个部分。在这部分中，我们需要描述数据处理方法，用于处理







由第一个部分“模型”产生的数据。这一部分实际上说明了我们是如何解决问题的。你必须有一个以上的解决方案。再提醒一遍：一个以上的解决方案。为了证明你有一个漂亮的方案，你需要有一个参照物，即一些可以与你的解决方案相比较的。你可以先从最简单，最常见的方案入手，然后逐步提炼，完善它，直到得到你的最好的解决方案。

一般情况下，对于离散的问题，最简单的解决方案可能就是随机选择。

在这一部分中，你需要证明你已经对问题进行了彻底的探讨，并且你已经尝试了许多不同的解决方案。即使你一开始就使用了最佳解决方案，然后尝试了一些其他的方案，在论文的书写中，你仍然应该表示从最根本的解决方案入手，然后逐步细化，最终达到你的最佳解决方案。如果你尝试了更先进的算法，但它的效率并不理想？也要把它放在论文中！用来表示你已经从不同的角度进行了尝试，即使你最好的解决方案并不是最复杂、最有趣的一个。在现实生活中，情况往往就是这样！

## 2.4.6 方案的比较 (Solution Comparison Methods)

有的时候，问题中会清楚地描述目标要求，以便于你构建算法的验证方法。比如2003MCM 问题 B “伽玛刀治疗方案”。题目中明确要求给出一种伽玛刀治疗方案来切除至少 90% 的肿瘤部分。这里你需要做的只是将所有解决方案的结果与 90% 作简单的比较，看看差别有多大。

然而，数学建模中还经常需要做一些决定来确定如何比较不同方案的结果。比如对于2014MCM 问题 A “除非超车否则靠右行驶的交通规则”。对于不同的规则，会得到不同的流量和安全性。这时你需要来确定一种权衡流量和安全的方法，来比较不同交通规则优劣。

对于很多问题来说，会有很多方法来比较不同的解决方案，最好用多种方法来评价它们。评价方法应该由大家一起自由讨论。

## 2.4.7 结果 (Results)

在这里，你需要表述测试结果。这一部分应该被特别关注，因为你已经将论文的其他部分表述完成了。如果可能的话，你可以提供大量的数据来支持你的结论。你的模型是不是将不同类型的数据集进行了整合？你的算法是如何做的？一般来说，这一部分将会以一些用到的参数结尾，这些参数出现在模型、解决方案和测试方法中。你应该尝试尽可能大的参数空间。在这一部分你要证明你已经采用了一个成熟的方法来处理问题，并且你已经尽可能地考查了问题的所有方面。

具体数据的展示是比较困难的。提供一些图表是最好的手段。但如果你彻底探讨了模型，解决方案和测试方法中出现的每一个参数，你将会有大量的数据需要罗列。

你应该以表格的形式来罗列数据，但不要指望评委会看这些表格。你需要在表格下面写一段解释性的文本，指出数据总的发展趋势，异常情况和整体结果。

重要提示：许多参赛队仅仅建立了一个模型，提出了一种解决方案，运行了一个检测

方法，给出了结果后就结束了。你必须要进行多次地测试！你必须要确定你的解决方案是稳定的！它可以适应一些微小的环境变化，你可以给参数一个微小的变化，调试你的代码，使它依然能返回正确的结果。让评委看到你的解决方案是灵活和稳定的，或者诚实地承认你的算法在一些特殊的情况下不能使用。这样你的论文会显得非常的全面。

### 2.4.8 结论-模型评价-改进方案(Conclusions-S&W-FutureWork)

首先，提出你的基本结论，即使你已经在上一个部分中提出过。如：“从整体上看，解决方案 A 的执行效率优于解决方案 B34%，优于解决方案 C67%”。

你需要用一些数字来概括所有的事情，提出一种方法来对数据进行某种平均并从中提取出少量的关键数字来对算法优劣排名。如果在结果里，你已经提到“算法 A 整体上看优于算法 B，而算法 B 也有自己的一些优点。”在结论部分中，你要摒弃前面的说法，直接说“A 是最好的”，这也需要放在摘要当中，表明你已经得到了具体，全面的结论。

模型评价这一部分是解释算法好的地方和需要改进地方的一个比较好的途径。推荐用项目符号列表。除了概括性的文字以外，不用过多的解释优缺点，结果部分中的主要观点也要在这里提及，同时提到缺点，以及任何限制性的假设。

为了证明你处理问题的方法是成熟的，可以提出改进方案。如果可以花数月时间继续研究这个问题，你会做些什么工作？是不是还有一些你想到的非常棒的算法，但还没有来得及在计算机上实现？竞赛是有时间限制，所以这个地方可以显示你对问题的一个整体把握。

### 2.4.9 参考文献 (References)

不要忘记引用任何你用到的参考资料。你们将会花很多时间在书中和网页上搜寻有用的东西：每次你找到有价值的东西时要做些简单的笔记。虽然有时你会想出一些原创的点子，但大多情况下你要做的是改进现有的模型和想法来使其适应当前的问题。在你的论文中，你必须在引用到文献的地方加上类似于“[1]”的标记，并在论文的末尾列出所有参考文献。

当你用到别人的想法或模型，却没有列在参考文献中，这叫剽窃，这是一个非常严重的问题。即使你没有抄文字，只是想法，那也算剽窃。如果评委们认为你的论文中涉嫌剽窃，那后果可不仅仅是你的论文获不了奖。需要一提的是，2007 年美国大学生数学建模竞赛中，中国某高校参赛队的作品在初审时被评为特等奖，后因涉嫌剽窃，取消了参赛资格，并给予通告。

### 2.4.10 论文的结构

首先需要注意的是，评委需要在短时间内评阅大量的论文。在论文初审时（这一步将淘汰一半数量的论文，这些论文被评为“successful participant”），对于每篇论文的评定，评委只有 5 分钟的时间。这意味着初审时评委只会阅读论文的摘要，并粗略的浏览一下全







文，然后将论文分成两堆。就这样，一半的论文被淘汰，并且再无翻身的可能。更糟糕的是，在第二轮的评审中，评审你论文的时间仍然只有五分钟左右。只有经过前两轮淘汰，你的论文才会被仔细阅读。这就意味着你的首要目标是顺利通过两个五分钟的评审。

除了摘要（文章最重要的部分），比较容易得到评委注意的部分还包括各小节标题，项目符号列表，表格，图形和数字。不要出现大的，不间断的文本，它们会使文章变得枯燥无味，而且可能永远都不能被完全读完。你应该使文字清楚，易读，文本应该被规则性地断开，通过标题，列表，数字，图表，以及任何你能想到的，可以使论文变成有趣的东西。

### 1. 小节的标题（Section）

每个小节的标题都是非常重要的。如果你去掉文章中的所有正文，各小节标题读起来就像一个大綱。评委是会详细看标题的，通过它评委可以了解到文章的流程（它应该和摘要中提到的整体思路相一致）。你至少应该安排两层的标题（标题，子标题，即二级标题），这会很容易把你的文章分成很多小块，每一块都会有明确的作用和目标。尽量不要出现一段或者两段没有标题的情况。它不仅可以使你的论文适合略读，还有利于明确文章主题，防止走题。例如以下内容：

- Summary
- Introduction
  - Assumptions
- TheModels
  - CoordinateSystemsandDefinitions
- InterpolationAlgorithms
  - Method1–Proximity
  - Method2–DensityMean
  - Method3–TrilinearInterpolation
  - Method4–PolynomialInterpolation
  - Method5–HybridAlgorithms
- TestingandResults
- StrengthsandWeaknesses
- FutureWork
- References

因此，最好给你的数学建模论文做一个目录，把所有小节的标题都列在一页纸上，方便评委快速了解你论文的结构。

### 2. 项目符号列表

为了清楚的显示一些要点，可以将这些要点用项目符号列表的方式呈现出来，例如以下两种项目符号列表：



- They break up blocks of texts, making reading less tedious. (1) 打破了大块文本, 减少了阅读的乏味感。
  - They emphasize important ideas. (2) 强调重要的思想。
  - They are easily noticed when skimming. (3) 当略读的时候, 容易得到关注。
- 读者可根据需要选用有无编号的项目符号列表。

### 3. 表格 (Tables) 和图 (Figures)

如果你编写了一个能够正常运行的计算机程序, 不要浪费它! 输入不同的参数值, 运行多次。然后以图 (如果你能) 或者表格的形式组织并展示数据。对于图表, 即使评委不加以细读, 也能留下深刻的印象。图表可以证明你有大量的数据来支持你的结论, 你已经对问题中出现的参数进行了彻底的探讨。

一张图胜过千言万语。图在模型部分非常有用, 可以展示你是如何处理问题的, 图永远是显示数据的最好方式。

## 2.5 论文排版

美国大学生数学建模竞赛必须同时提交纸质版和电子版两种形式的论文, 电子版通常为 AdobePDF 文档或者 MS Word 文档。论文的初审阶段, 评委只花几分钟来评审一篇论文, 论文的版式是论文的门面, 因此论文排版的美观程度也影响论文的评审。但这并不意味着花费大量时间来排版是值得的, 笔者认为选择 LaTeX 或 MS Word 都可以, 只要排出来的论文不影响阅读就可以。

MS Word 相对比较简单, 是“所见即所得”的文字排版软件。绝大多数使用 MS Word 的经验, 本书对于 MS Word 的排版知识不作介绍。

LaTeX 是一种基于 TEX 的排版系统, 由美国计算机学家 Leslie Lamport 在 20 世纪 80 年代初期开发, LaTeX 非常适用于生成高印刷质量的科技和数学类文档, 能够生成复杂表格和数学公式。与 MS Word 不同, LaTeX 不是“所见即所得”的文字排版软件。LaTeX 是一种标识语言, 可配置性极高, 短时间系统的学习 LaTeX 并不太可能。不过, 书写 MCM 论文的话最基本的配置和命令就足够了。网络上有一些开源的 MCM 论文模版<sup>[3,4]</sup>, 读者可以基于这些模版快速的排出精美的 MCM 论文。由于 LaTeX 能够生成排版精美的 PDF 文档, 因此很多指导老师都推荐使用 LaTeX 来书写 MCM 论文。本节介绍 LaTeX 基本命令, 读者如果希望了解更多 LaTeX 相关知识, 可以参考“一份不太简短的 LaTeX2 介绍”<sup>[5]</sup>。

### 2.5.1 LaTeX 软件的安装和使用

LaTeX 文档是扩展名为 .tex 的纯文本文档, 比如 main.tex。建立 LaTeX 文档可以使用文档编辑软件, 也可以使用文字处理软件并保存为纯文本格式。LaTeX 文档必须经过编译才能生成 PDF 文档。





推荐 Windows 中文读者使用 CTEX 软件（包括文档编辑软件 WinEdt），CTEX 软件支持中英两种语言，可以从 <http://www.ctex.org/> 免费获得。WinEdt 中可通过工具栏内点击 TeX 选项，然后点击 PDF 选项，再点击 PDFLaTeX 选项来由 TEX 文件编译生成 PDF。如果文档含有交叉索引，则需要将文档编译两次才能获得正确的索引。Linux 用户可使用 TexLive，TexLive 可以从 <https://www.tug.org/texlive/> 免费获得。Linux 用户可以通过 pdfLaTeX 命令在终端进行编译。

LaTeX 命令的格式是：以反斜杠(\)作为标示符，其后是指令，再后面是包含在花括号({})中的参数。以百分号(%)开头的行为注释行，不参与编译。

## 2.5.2 简单示例

在 LaTeX 中首先要知道的命令是 \documentclass。 \documentclass 命令用于指定文档的类型，文档的类型可以为 article, book, beamer 等，其基本用法如下：

`\documentclass[选项]{文档类型}`

下面给出一个简单的文档例子，排版效果如图 2-6 所示，具体命令如下：

```
\documentclass[a4paper,11pt]{article}
\usepackage{amsmath, amssymb}
\title{\LaTeX}
\author{Joe}
\date{January 25, 2015}
\begin{document}
\maketitle
\LaTeX{} is a document preparation
system for the \TeX{} typesetting
program.
\end{document}
```

LaTeX  
Joe  
January 25, 2015

LaTeX is a document preparation system for the TeX typesetting program.

图 2-6

命令 \usepackage 用于载入宏包，amsmath 及 amssymb 宏包提供了数学排版的许多功能以及各种数学符号。命令 \author 用于指定文档的作者，\title 用于指定文档的标题。 \date 命令用于指定日期。 \begin{document} 和 \end{document} 用于指定文档正文的开始和结束。 \maketitle 用来生成文章标题，作者和日期。

## 2.5.3 小节生成

文档一般都有多个层次结构，在 LaTeX 中 \section, \subsection, \subsubsection 可生成不同层次的结构。这些命令会自动生成小节标题。下面给出一个简单的例子。排版效果如图 2-7 所示，具体命令如下：

```
\subsection{subsection}\label{sec:
sub}
This is the subsection \ref{sec: sub}.
```

### 6.4 subsection

This is the subsection 6.4.

图 2-7



以上例子中\label 命令用于标记,\ref 用于引用。这种交叉引用的方式同样适用于图表,这在我们后面的例子中会涉及到。

### 2.5.4 公式输入

LaTeX 中的数学表达式必须在数学模式下输入,同行排列的数学表达式在两个美元符号(\$)之间,独立成行的数学表达式在\[和\]之间,如果要使独立成行的数学表达式自动带有编号,则数学表达式应在\begin{equation}和\end{equation}之间,下面给出几个公式的例子,排版效果如图 2-8 所示,具体命令如下:

```
A inline equation is shown as $E = m \cdot c^2$, a display equation
with number is shown as equation
(\ref{eq:EMC})
\begin{equation}\label{eq:EMC}
E = m \cdot c^2
\end{equation}
and a display equation without
number is shown as follow
\[ E=m \cdot c^2 \]
```

A inline equation is shown as  $E = m \cdot c^2$ , a display equation with number is shown as equation

$$E = m \cdot c^2$$

and a display equation without number is shown as follow

$$E = m \cdot c^2$$

图 2-8

### 2.5.5 项目符号列表

为了清楚地显示一些要点,可以将一些要点用项目符号列表的方式呈现出来。LaTeX 中的 itemize 环境和 enumerate 环境是两种最常用的项目符号列表。下面给出两种项目符号列表的例子,排版效果如图 2-9 所示,具体命令如下:

```
Itemize 环境:
\begin{itemize}
\item 观点 1
\item 观点 2
\item 观点 3
\end{itemize}
Enumerate 环境:
\begin{enumerate}
\item 情况 1
\item 情况 2
\item 情况 3
\end{enumerate}
```

Itemize 环境:

- 观点 1
- 观点 2
- 观点 3

Enumerate 环境:

- 情况 1
- 情况 2
- 情况 3

图 2-9

### 2.5.6 图片插入

LaTeX 中插入图片,需要在 figure 环境下用\includegraphics。如果想并列排图,可以借助 minipage 环境。下面给出插入图片的实例。排版效果如图 2-10 所示,具体命令如下:



```

一个图例见图\ref{fig:01}, 两图并排见图\ref{fig:02}和图\ref{fig:03}
\begin{figure}[!htb]
\centering
\includegraphics[width=0.6\textwidth]{fig01.pdf}
\caption{图 1 标题}\label{fig:01}
\end{figure}

\begin{figure}[!htb]
\begin{minipage}[t]{0.5\linewidth}
\centering
\includegraphics[width=0.8\textwidth]{fig02.pdf}
\caption{图 2 标题}\label{fig:02}
\end{minipage}
\begin{minipage}[t]{0.5\linewidth}
\centering
\includegraphics[width=0.8\textwidth]{fig03.pdf}
\caption{图 3 标题}\label{fig:03}
\end{minipage}
\end{figure}

```

一个图例见图 1, 两图并排见图 2 和图 3

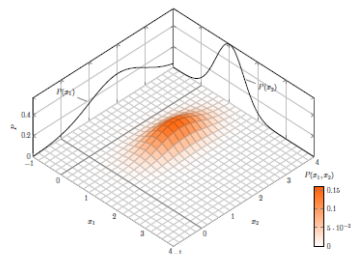


图 1 标题

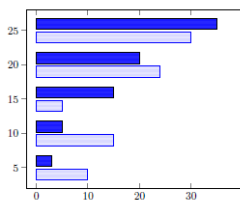


图 2 标题

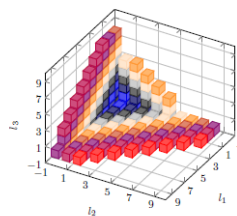


图 3 标题

图 2-10

以上实例中`\centering`用于使图片居中, `\caption`用于生成图片的标题。

## 2.5.7 表格插入

LaTeX 中插入表格, 需要在 `table` 环境中, 下面给出插入表格的实例, 排版效果如图 2-11 所示, 具体命令如下:

```

见表\ref{tab:name}.
\begin{table}[!htb]
\centering
\caption{Table
example}\label{tab:name}
\begin{tabular}{|c|c|c|c|c|}
\hline
& A & B & C & D \\ \hline
X & 1 & 2 & 3 & 4 \\ \hline
Y & 5 & 6 & 7 & 8 \\ \hline
\end{tabular}
\end{table}

```

见表 1.

表 1: Table example

	A	B	C	D
X	1	2	3	4
Y	5	6	7	8

图 2-11

## 2.5.8 引用文献

LaTeX 中可以使用 BibTeX 来管理参考文献, BibTeX 使用数据库的方式来管理参考文献. BibTeX 文件的后缀名为 bib。先来看一个例子. 在 biblo.bib 文件中, 一条参考文献对应一个记录, 由一个唯一的 ID 描述, 下面给出两条参考文献的记录:

```
@article{IDname01,
author = {作者, 多个作者用 and 连接},
title = {标题},
journal = {期刊名},
volume = {卷 20},
number = {页码},
year = {年份}

@book{ IDname02,
author ={作者},
year={年份 2008},
title={书名},
publisher ={出版社名称}
```

第一行 @article 告诉 BibTeX 这是一个文章类型的参考文献。还有其他格式, 例如 book, booklet, conference, inbook, incollection, inproceedings, manual, misc, mastersthesis, phdthesis, proceedings, techreport, unpublished 等等。接下来的 “IDname01”, 就是你在正文中应用这个条目的名称, 这个名称可以随便取, 但每个文献的名称必需唯一。为了在 LaTeX 中使用 BibTeX 数据库, 必须先做下面三件事情。

(1) 设置参考文献的类型 (bibliography style)。标准的为 plain:

```
\bibliographystyle{plain}
```

将上面的命令放在 LaTeX 文档的 \begin{document} 后边。

(2) 标记引用 (Make citations)。当你在文档中想使用引用时, 插入 LaTeX 命令


```
\cite{引用文章名称}
```

“引用文章名称” 就是前边定义在 @article 后面的名称。

(3) 告诉 LaTeX 生成参考文献列表。在 LaTeX 文档结束前输入:

```
\bibliography{bibfile}
```

这里 bibfile 就是你的 BibTeX 数据库文件 biblo.bib。

此时在编译 TEX 文档时, 需要先用 LaTeX 编译 .tex 文件, 生成一个 .aux 的文件, 再用 BibTeX 编译 .bib 文件, 再次用 LaTeX 编译 .tex 文件, 这个时候在文档中已经包含了参考文献, 但此时引用的编号可能不正确, 最后用 LaTeX 编译 .tex 文件, 如果一切顺利的话, 这时所有东西都已正常了。值得注意的是 .bib 文件需和 .tex 文件放于同一目录下。 







## 第3章 软件快速入门

### 3.1 MatLab快速入门

#### 3.1.1 引言

MatLab 是 Matrix Laboratory “矩阵实验室”的缩写。MatLab 语言是由美国的 Clever Moler 博士于 1980 年开发的，初衷是为了解决“线性代数”课程的矩阵运算问题。1984 年由美国 Math Works 公司推向市场，历经十多年的发展与竞争，现已成为国际公认的最优秀的工程应用开发环境。MatLab 功能强大、简单易学、编程效率高，深受广大科技工作者的欢迎。

在数学建模竞赛中，由于只有短短的三到四天，而论文的评判不仅注重计算的结果更注重模型的创造性等很多方面，因此比赛中把大量的时间花费在编写和调试程序上只会喧宾夺主，是很不值得的。使用 MatLab 可以很大程度上的方便计算、节省时间，使我们精力更多的放在模型的完善上，所以是较为理想的。

这里快速地介绍一下 MatLab 与数学建模相关的基础知识，并列举一些简单的例子，很多例子都是源于国内外的数学建模赛题。希望能帮助同学们在短时间内方便、快捷的使用 MatLab 解决数学建模中的问题。当然要想学好 MatLab 更多地依赖自主学习，一个很好的学习 MatLab 的方法是查看 MatLab 的帮助文档：

如果你知道一个函数名，想了解它的用法，你可以用 ‘help’ 命令得到它的帮助文档：

```
>>help 函数名
```

如果你了解含某个关键词的函数，你可以用 ‘lookfor’ 命令得到相关的函数：

```
>>lookfor 函数名
```

例如 helpsum 命令将输出 sum 函数的帮助信息.其他一些可能有用的帮助命令 ‘info’，‘what’ 和 ‘which’ 等。这些命令的详细用法和作用都可以用 help 获得. MatLab 中还提供了很多程序演示实例，这些例子可以通过 ‘demo’ 获得。





### 3.1.2 变量

MatLab 程序的基本数据单元是数组，一个数组是以行和列组织起来的数据集合，并且拥有一个数组名。标量在 MatLab 中也被当作数组来处理，它被看作只有一行一列的数组。数组可以定义为向量或矩阵。向量一般来描述一维数组，而矩阵往往来描述二维或多维数组。数组中的任何元素都可以是实数或者复数；在 MatLab 中， $\sqrt{-1}$  是由 'i' 或 'j' 来表示，当然前提是用户没有预先重新定义 'i' 或 'j'。MatLab 中，数组的定义要用 '[' 来括起来，数组中同一行元素间以空格或逗号 ',' 隔开，行与行之间由分号 ';' 隔开。下面给出实数，复数，行向量，列向量和矩阵的定义及赋值方式。

```
实数      >> x = 5
复数      >> x = 5 + 10i 或者 >> 5 + 10j
行向量    >> x = [1 2 3] 或者 >> x = [1, 2, 3]
列向量    >> x = [1; 2; 3]
3×3矩阵   >> x = [1 2 3; 4 5 6; 7 8 9]
```

需要注意的是，一个数组每一行元素的个数必须完全相同，每一列元素的个数也必须完全相同。对于复数的输入，虚部前的系数和 'i' 或 'j' 之间不能有空格，如  $-1+2i$  是不对的，而  $-1+2i$  或  $-1+ix2$  才是有效的输入方式。

#### 1. 固定变量

前面我们说，如果用户没有预先重新定义 'i' 或 'j'，则 'i' 或 'j' 表示  $\sqrt{-1}$ 。在 MatLab 中还有几个常见的固定变量，如果用户没有预先重新定义，这些固定的变量有着自身的意义。

```
pi         $\pi$ 
i, j       $\sqrt{-1}$ 
inf        $\infty$ 
ans       默认变量
```

NaN 是 notanumber 的缩写，像  $0/0$ ， $\text{inf}*\text{inf}$  等情况的计算会产生非数；ans 是 answer 的缩写，如果不定义变量，MatLab 会将运算结果放在默认变量 ans 中。

#### 2. 复数运算

MatLab 中提供了一些复数运算的函数，这里列出一些重要的复数运算函数。

```
复数输入    >> x=3+4j
实部        >> real(x) ⇒ 3
虚部        >> image(x) ⇒ 4
模长        >> abs(x) ⇒ 5
共轭复数    >> conj(x) ⇒ 3-4i
幅角        >> angle(x) ⇒ 0.9273
```



### 3. 向量，矩阵的快捷生成

创建一个小数组用一一列举出元素的方法是比较容易的，但是当创建包括成千上万个元素的数组时则不太现实。在 MatLab 中，向量可以通过冒号‘:’方便快捷地生成，用两个冒号按顺序隔开‘第一个值’，‘步增’和‘最后一个值’就可生成指定的向量。如果步增为 1，则可以省略掉步增和一个冒号，比如：

```
>>x=1: 0.5: 3⇒ [1.01.52.02.53.0]
```

```
>>y=1: 3⇒ [123]
```

向量的快捷生成还可以调用函数 `linspace`，`linspace` 函数只需给出向量的第一个值，最后一个值和等分的个数就可以生成指定的向量。转置运算符（单引号）‘`’`可以用来将行向量转置为列向量，或更加复杂的矩阵的转置。

```
>>x=[1: 2: 5]')x=[1;3;5]
```

对于特殊的向量和矩阵，MatLab 提供了一些内置函数来创建它们。例如，函数 `zeros` 可以初始化任何大小的全为零的数组。如果这个函数的参数只是一个标量，将会创建一个方阵，行数和列数均为这个参数。如果这个函数有两个标量参数，那么第一个参数代表行数，第二个参数代表列数。

```
>>x=zeros(2)=)x=[0 0 ; 0 0]
```

```
>>x=zeros(2,3)=)x=[0 0 0 ; 0 0 0]
```

类似的内置函数还有 `ones`，`eye`，用法与 `zeros` 一致。函数 `ones` 产生的数组包含的元素全为 1；函数 `ones` 用来产生单位矩阵，只有主对角线的元素为 1。

### 4. 获取向量/矩阵中的元素

通过定元素所在的指行和列，可以获得矩阵中指定的一个或多个元素。比如可以用以下语句获得矩阵 `A=[1 2 3; 4 5 6; 7 8 9]` 的第 1 行的第 3 列的元素

```
>>x=A(1,3)=)x=3
```

可以用以下语句获得矩阵 `A` 的第 2 行所有元素

```
>>y=A(2,:)=)x=[4 5 6]
```

其中的冒号‘:’表示“所有列”的意思。`A` 矩阵的前两行前两列组成的矩阵可通过以下语句获得：

```
>>z=A(1:2,1:2)=)z=[1 2 ; 4 5]
```

## 3.1.3 矩阵和数组运算

针对矩阵和数组，MatLab 有一系列的算术运算，关系运算和逻辑运算。

### 1. 算术运算

矩阵的基本数学算数主要有加法，减法，乘法，右除，左除，指数和转置运算，运算符号如下：





+加法运算  
-减法运算  
×乘法运算  
/右除运算  
\左除运算  
^指数运算  
'转置运算

这些运算的法则都与线性代数中的相同。如果矩阵  $A$  与矩阵  $B$  相乘, 则必需满足  $A$  的列数必需等于  $B$  的行数, 否则 MatLab 会报错。对于左除和右除运算,  $x=A \backslash b$  是  $Ax=b$  的解, 而  $x=b/A$  是  $xA=b$  的解。

上面介绍的矩阵运算中的乘除运算都不是针对同阶数组对应分量的运算, 而 MatLab 中还有针对同阶数组对应分量的运算, 这种称为数组的运算, 也称点运算。点算包括点乘, 点除和点乘方。

. ×乘法运算  
./右除运算  
\左除运算  
.^指数运算

以下给出一个例子来说明矩阵运算和数组运算的区别:

```
>>A=[1 2; 3 4]
```

```
A=
```

```
1      2  
3      4
```

```
>>B=A×A
```

```
B=
```

```
7      10  
15     22
```

```
>>C=A.×A
```

```
C=
```

```
1      4  
9      16
```

## 2. 关系运算

关系运算用来判断两同阶数组 (或者一个是矩阵, 另一个是标量) 对应分量的间的大小关系。关系运算包括以下操作:

<小于  
<=小于等于  
>大于



$\geq$  大于等于

$==$  等于

$\sim$  不等于

若参与运算的是两个矩阵，关系运算是将两个矩阵对应元素逐一进行关系运算，关系运算的结果是一个同维数逻辑矩阵，其元素值为只含 0（假）和 1（真）。若参与运算的一个是矩阵，另一个是标量，则是矩阵中每个元素与该标量进行关系运算，最终产生一个同维数逻辑矩阵，其元素值为只含 0（假）和 1（真）。例如

```
>>B=[2 1 3 5 4]
>>C=A>B%C=[1>2,3>1,4>3,2>5,5>4]
C=
0 1 1 0 1
>>D=A<=3%D=[1<=3,3<=3,4<=3,2<=3,5<=3]
D=
1 1 0 1 0
```

需要注意的是 $=$ 和 $==$ 差别：前者是赋值运算；后者是关系运算符“等于”，判断是否相等。

### 3. 逻辑运算

MatLab 的基本逻辑运算符为： $\&$ （与）， $|$ （或）， $\sim$ （非）。参与逻辑运算的是两个同维数矩阵；或者一个是矩阵，另一个是标量。若参与运算的是两个矩阵，逻辑运算是将两个矩阵对应元素逐一进行逻辑运算，逻辑运算的结果是一个同维数逻辑矩阵，其元素值为只含 0（假）和 1（真）。若参与运算的一个是矩阵，另一个是标量，则是矩阵中每个元素与该标量进行逻辑运算，最终产生一个同维数逻辑矩阵，其元素值为只含 0（假）和 1（真）。例如

```
>>A=[1 3 4 2 5];
>>B=[2 1 3 5 4];
>>C=(A>B)&(A<=3)
C=
0 1 0 0 0
```

逻辑运算常与 MatLab 中的程序控制结构语句（如 if，while 等）相结合。

### 4. 常用数学函数

MatLab 中包括了大量的内置函数，这些函数的组合可以很方便的完成很多复杂的功能。下面给出常用的数学函数：

sin 正弦	asin 反正弦
cos 余弦	acos 反余弦
tan 正切	atan 反正切
cot 余切	acot 反余切





exp 指数函数	sqrt 平方根
log 自然对数	log10 以 10 为底的对数
abs 绝对数	sign 符号函数
min 取小	max 取大
sum 求和	

以上这些函数大多（除 min, max 和 sum）也是针对矩阵对应元素逐一进行函数的运算，比如

```
>>theta=0:pi/3:pi
theta=
0 1.0472 2.0944 3.1416
>> sin(theta)
ans =
0 0.8660 0.8660 0.0000
```

### 3.1.4 控制结构语句

在 MatLab 中，常用的控制结构语句主要是 if 结构和 for 结构。if 语句有 3 种格式：

（1）单分支 if 语句，其格式如下。

当条件成立时，则执行语句组，执行完之后继续执行 if 语句的后继语句，若条件不成立，则直接执行 if 语句的后继语句。

```
if 条件
语句
end
```

（2）双分支 if 语句，其格式如下。

当条件成立时，执行语句组 1，否则执行语句组 2，语句组 1 或语句组 2 执行后，再执行 if 语句的后继语句。

```
if 条件
语句组 1
else
语句组 2
end
```

（3）多分支 if 语句。

其格式如下。多分支 if 语句用于实现多分支选择结构，可以代替 switch 语句。

```
if 条件 1
语句组 1
```



```
elseif 条件 2
语句组 2
.....
elseif 条件 n
语句组 n
else
语句组 n+1
end
```

循环结构 for 语句的格式如下：

```
for 循环变量=表达式 1: 表达式 2: 表达式 3
循环体语句
end
```

其中表达式 1 的值为循环变量的初值，表达式 2 的值为步长，表达式 3 的值为循环变量的终值。步长为 1 时，表达式 2 可以省略。“表达式 1: 表达式 2: 表达式 3”还可利用向量或矩阵替代，执行过程是依次将矩阵的各列元素赋给循环变量，然后执行循环体语句，直至各列元素处理完毕。下面给出一个例子来说明 if 和 for 语句的用法：

```
%计算 10 以内的奇数和
tot=0;
for i=1: 10
    if mod(i,2);tot=tot+i;end
end
```

在 MatLab 编程中，采用循环语句会降低其执行速度，因此如果可以用内置函数或数组或矩阵的运算实现的功能，尽量不用循环语句，例如以上程序可以由以下语句实现：

```
tot=sum(1:2:10);
```

其他控制结构语句还有 while, switch, continue, break, return 等，请读者自己了解这些控制结构语句。

### 3.1.5 MatLab文件

为了保存程序，数据，图像等。MatLab 有一系列的文件来存储程序，数据，图像等。本节主要介绍用于保存 MatLab 程序脚本的 m 文件和 m 函数文件以及 fig 图形文件。

#### 1. M 文件和 M 函数文件

当用户要运行的指令较多时，直接从键盘上逐行输入指令比较麻烦，而命令文件可以较好地解决这一问题。用户可以将一组相关指令编辑在同一个文件中，即从指令窗口工具







栏的新建按钮或选择菜单 File: New: M-File 进入 MatLab 的程序编辑器窗口, 以编写自己的 M 文件, 运行 M 文件时, 只需点击工具栏中的运行按钮或在命令窗口中输入文件名字, MatLab 就会自动按顺序执行文件中的命令。如图 3-1 所示, 一个文件名为 main.m 的 M 文件。需要注意的是:

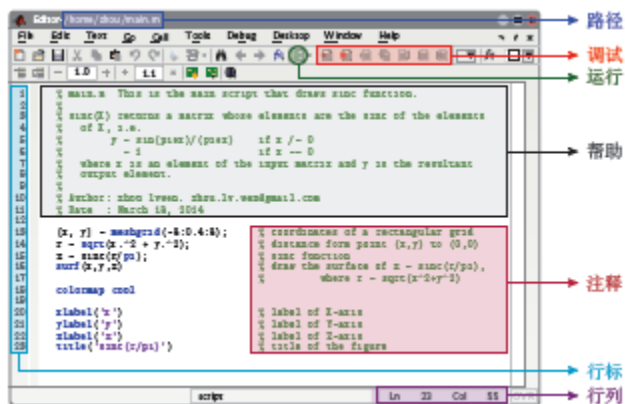


图 3-1

- M 文件名只能是字母, 下划线和数字的组合, 且只能以字母开头。M 文件名也不要与 MatLab 内置函数名相同。
- 在 MatLab 中, “%”为注释符, %后的语句用于注释, 不会被执行。
- 以“;”结尾的 MatLab 命令语句, 将不会在命令窗口中输出该语句的运行结果。

M 文件中有一种特殊的文件, 这种文件除注释外以 function 开头, 我们称这样的文件为 M 函数文件。M 函数文件用得更为广泛, 它具有参数传递功能, 可以很方便地在命令窗口或其他脚本文件中进行调用。M 函数文件的文件格式如下:

function[输出 1,输出 2, ……]=函数名(输入 1,输入 2,……)

%函数的一些说明

MatLab 语句 1;

……

MatLab 语句 n;

需要注意的是: 函数文件的变量是局部变量, 运行期间有效, 运行完毕就自动被清除; M 函数文件名必须与函数名一致, 即 M 函数文件必需保存成“函数名.m”。下面给出一个 M 函数的例子, 该 M 函数用来把角度转化为弧度的函数, M 函数名为“ang2rad.m”。

```
function rad=ang2rad(ang)
```

%ANG2RAD: 这是一个把角度转化为弧度的函数

```
rad=ang/180*pi;
```

有了这个函数, 我们就可以在需要把角度转化为弧度的地方直接调用这个 M 函数, 比如

```
>>theta=0:60:180
```



```
theta=
0 60 120 180
>>ang2rad(theta)
ans=
0 1.0472 2.0944 3.1416
```

## 2. FIG 文件

MatLab 最强大的功能之一是作图，用 MatLab 作图后，可以把图存成 FIG 文件。图 3-2 所示是 MatLab 作图后显示出来的图形窗口。通过 File: Save 可以把图存成“图名.fig”的文件。FIG 文件保存了图形的所有信息，可以方便以后的编辑，以及转存为其他格式的图片。如果有需要，我们也可以从 FIG 文件中获得图形的相关数据。

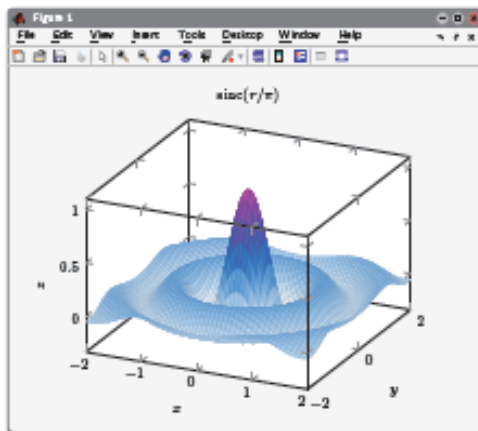


图 3-2

## 3.1.6 作图

MatLab 最强大的功能之一是作图，MatLab 中包括了大量的二维和三维的作图函数。本节我们简单的介绍一下 MatLab 二维作图命令及三维作图命令。

### 1. 二维作图命令

MatLab 中最常用也是最基本的二维作图命令为 plot，plot 函数针对向量或矩阵的列来绘制曲线的。调用 plot 函数的常用格式有以下几种：

- plot(x): 当 x 为一向量时，以 x 元素的值为纵坐标，x 的序号为横坐标值绘制曲线。  
当 x 为一实矩阵时，则以其序号为横坐标，按列绘制每列元素值相对于其序号的曲线，当 x 为  $m \times n$  矩阵时，就有 n 条曲线。
- plot(x, y): 以 x 元素为横坐标值，y 元素为纵坐标值绘制曲线。
- plot(x, y1, x, y2, ...): 以公共的 x 元素为横坐标值，以 y1, y2... 元素为纵坐标值绘制



多条曲线。

值得注意的是，在上面的每种用法中，还可额外指定颜色，点形或线形来区分不同的数据组。下面给出一个简单的实例，如图 3-3 所示。

```
%画 sin 和 cos 曲线
x=-2*pi:0.1:2*pi;
y1=sin(x);
y2=cos(x);
plot(x, y1, '-b');
hold on
plot(x, y2, '-r');
xlabel(' x ')
ylabel('y')
text(0,0,'(0,0)')
legend('sin(x)', 'cos(x)')
')
```

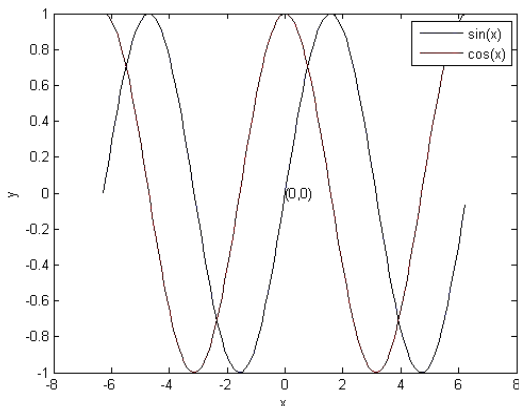


图 3-3

以上程序中 `plot(x,y1,'-b')` 画出了一箱  $y=\sin x$  的曲线，其中参数 `'-b'` 指定了曲线为蓝色实线。更多参数选项见表 3-1 所示。在上面的程序中还用用到了一些常用命令，下面更多常用的命令。

<code>xlabel</code>	$x$ 坐标轴标签
<code>ylabel</code>	$y$ 坐标轴标签
<code>title</code>	在图上方加标题
<code>grid on/grid off</code>	开启/关闭网格
<code>text</code>	在指定位置标文本
<code>axis</code>	控制 $x$ 坐标和 $y$ 坐标的范围
<code>hold on/hold off</code>	开启 / 关闭保持当前图形，允许多个图形叠加于同一坐标系。

表 3-1

颜 色		点 形				线 形	
b	蓝	.	点	^	向上三角形	-	实线
g	绿	o	圆	<	向左三角形	:	点线
r	红	x	叉号	>	向右三角形	-.	点划线
c	青	+	加号	p	五角星	--	虚线
m	紫	*	星号	h	六边形		
y	黄	s	正方形				
k	黑	d	菱形				
w	白	v	向下三角形				

除了 `plot` 外，还有一些常用的二维作图命令，下面给出这些命令的列表，见表 3-2 请读者自行查找帮助文档来了解它们的用法。

表 3-2

命 令	功 能
loglog	使用对数坐标系绘图
semilogx	横坐标为对数坐标轴
semilogy	横坐标为线性坐标轴
polar	极坐标图
bar	直方图
errorbar	误差棒图
pie	制饼图
hist	统计直方图

## 2. 三维作图命令

与二维曲线作图函数 `plot` 相对应，MatLab 提供了 `plot3` 函数，可以在三维空间中绘制三维曲线，它的格式类似于 `plot`，不过多了  $z$  方向的数据。这里不再详述其调用格式，给出一个实例图（图 3-4）供读者学习。

```
t=0:pi/50:10*pi;
x=sin(t);
y=cos(t);
z=t;
plot3(x,y,z,'b-');
title('Helix');
xlabel('sin(t)');
ylabel('cos(t)');
zlabel('t');
text(0,0,'Origin');
grid on
```

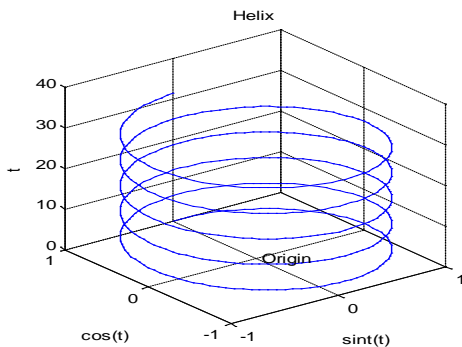


图 3-4

除了三维空间曲线，三维作图中还有三维曲面作图。用于绘制三维曲面的函数主要有 `mesh` 和 `surf`。在用 `mesh` 和 `surf` 绘制三维曲面时，常伴随着函数 `meshgrid` 的使用，`meshgrid` 是 MatLab 中用于生成网格采样点的函数，生成绘制 3-D 图形所需的网格数据。下面给出 `surf` 绘制曲面的实例。MatLab 中用于生成网格采样点的函数，生成绘制 3-D 图形所需的网格数据。下面给出 `surf` 绘制曲面的实例，如图 3-5 所示。

```
%画曲面 z=sin(x)*cos(y)
[x,y]=meshgrid(-pi:0.1:pi);
z=sin(x).*cos(y);
surf(x,y,z)
xlabel('x')
ylabel('y')
zlabel('z')
title('sinx siny')
```

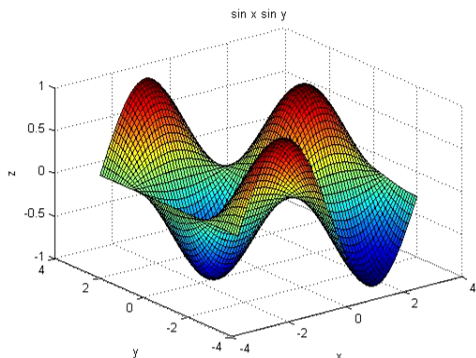


图 3-5



以上程序，先用 `meshgrid` 函数产生在  $x$ - $y$  平面上的二维的网格数据，再以一组  $z$  轴的数据对应到这个二维的网格，即可用 `surf(x,y,z)` 画出三维的曲面。

## 3.2 Lingo入门

### 3.2.1 Lingo基础知识讲解

#### 1. Lingo 基本概念

LINGO 是 Linear Interactive and General Optimizer 的缩写，即“交互式的线性 and 通用优化求解器”，由美国 LINDO 系统公司（Lindo System Inc.）推出的，可以用于求解非线性规划，也可以用于一些线性和非线性方程组的求解等，功能十分强大，是求解优化模型的最佳选择。其特色在于内置建模语言，提供十几个内部函数，可以允许决策变量是整数（即整数规划，包括 0-1 整数规划），方便灵活，而且执行速度非常快。能方便与 Excel，数据库等其他软件交换数据。

#### 2. Lingo 函数

Lingo 有 9 种类型的函数。

- ① 基本运算符：包括算术运算符、逻辑运算符和关系运算符。
- ② 数学函数：三角函数和常规的数学函数。
- ③ 金融函数：Lingo 提供的两种金融函数。
- ④ 概率函数：Lingo 提供了大量概率相关的函数。
- ⑤ 变量界定函数：这类函数用来定义变量的取值范围。
- ⑥ 集操作函数：这类函数为对集的操作提供帮助。
- ⑦ 集循环函数：遍历集的元素，执行一定的操作的函数。
- ⑧ 数据输入输出函数：这类函数允许模型和外部数据源相联系，进行数据的输入输出。
- ⑨ 辅助函数：各种杂类函数。

##### （1）基本运算符

这些运算符是非常基本的，甚至可以不认为它们是一类函数。事实上，在 Lingo 中它们是非常重要的。

算术运算符

算术运算符是针对数值进行操作的。Lingo 提供了 5 种二元运算符：

- ^ 乘方
- × 乘
- / 除



+ 加

- 减

Lingo 唯一的一元算术运算符是取反函数“-”。

这些运算符的优先级由高到底为：

高 -（取反）

^

× /

低 + -

运算符的运算次序为从左到右按优先级高低来执行。运算的次序可以用圆括号“（）”来改变。

### 逻辑运算符

在 Lingo 中，逻辑运算符主要用于集循环函数的条件表达式中，来控制在函数中哪些集成员被包含，哪些被排斥。在创建稀疏集时用在成员资格过滤器中。

Lingo 具有 9 种逻辑运算符：

#not# 否定该操作数的逻辑值，#not# 是一个一元运算符

#eq# 若两个运算数相等，则为 true；否则为 false

#ne# 若两个运算符不相等，则为 true；否则为 false

#gt# 若左边的运算符严格大于右边的运算符，则为 true；否则为 false

#ge# 若左边的运算符大于或等于右边的运算符，则为 true；否则为 false

#lt# 若左边的运算符严格小于右边的运算符，则为 true；否则为 false

#le# 若左边的运算符小于或等于右边的运算符，则为 true；否则为 false

#and# 仅当两个参数都为 true 时，结果为 true；否则为 false

#or# 仅当两个参数都为 false 时，结果为 false；否则为 true

这些运算符的优先级由高到低为：

高 #not#

#eq# #ne# #gt# #ge# #lt# #le#

低 #and# #or#

字母缩写辅助：g:greater, l:less, e:equal, t:than, n:not

### 关系运算符

在 Lingo 中，关系运算符主要是被用在模型中，来指定一个表达式的左边是否等于、小于等于或者大于等于右边，形成模型的一个约束条件。关系运算符与逻辑运算符#eq#、#le#、#ge#截然不同，前者是模型中该关系运算符所指定关系的为真描述，而后者仅仅判断一个该关系是否被满足：满足为真，不满足为假。

Lingo 有三种关系运算符：“=”、“<=”和“>=”。Lingo 中还能用“<”表示小于等于关系，“>”表示大于等于关系。Lingo 并不支持严格小于和严格大于关系运算符。然而，







如果需要严格小于和严格大于关系，比如让 A 严格小于 B：

$$A < B$$

那么可以把它变成如下的小于等于表达式：

$$A + \varepsilon \leq B$$

这里  $\varepsilon$  是一个小的正数，它的值依赖于模型中 A 小于 B 多少才算不等。

下面给出以上三类操作符的优先级：

高 #not# - （取反）

^

\* /

+ -

#eq# #ne# #gt# #ge# #lt# #le#

#and# #or#

低 <= = >=

## （2）数学函数

Lingo 提供了大量的标准数学函数：

@abs(x) 返回 x 的绝对值

@sin(x) 返回 x 的正弦值，x 采用弧度制

@cos(x) 返回 x 的余弦值

@tan(x) 返回 x 的正切值

@exp(x) 返回常数 e 的 x 次方

@log(x) 返回 x 的自然对数

@lgm(x) 返回 x 的 gamma 函数的自然对数

@sign(x) 如果 x < 0 返回 -1；否则，返回 1

@floor(x) 返回 x 的整数部分。当 x ≥ 0 时，返回不超过 x 的最大整数；当 x < 0 时，返回不低于 x 的最大整数。

@smax(x1, x2, ..., xn) 返回 x1, x2, ..., xn 中的最大值

@smin(x1, x2, ..., xn) 返回 x1, x2, ..., xn 中的最小值

## （3）概率函数

① @pbn(p, n, x)

二项分布的累积分布函数。当 n 和（或）x 不是整数时，用线性插值法进行计算。

② @pcx(n, x)

自由度为 n 的  $\chi^2$  分布的累积分布函数。

③ @peb(a, x)

当到达负荷为 a，服务系统有 x 个服务器且允许无穷排队时的 Erlang 繁忙概率。

④ @pel(a, x)

当到达负荷为 a，服务系统有 x 个服务器且不允许排队时的 Erlang 繁忙概率。

⑤ @pfd( $n, d, x$ )

自由度为  $n$  和  $d$  的  $F$  分布的累积分布函数。

⑥ @pfs( $a, x, c$ )

当负荷上限为  $a$ ，顾客数为  $c$ ，平行服务器数量为  $x$  时，有限源的 Poisson 服务系统的等待或返修顾客数的期望值。 $a$  是顾客数乘以平均服务时间，再除以平均返修时间。当  $c$  和（或） $x$  不是整数时，采用线性插值进行计算。

⑦ @phg(pop,  $g, n, x$ )

超几何 (Hypergeometric) 分布的累积分布函数。pop 表示产品总数， $g$  是正品数。从所有产品中任意取出  $n$  ( $n \leq \text{pop}$ ) 件。pop,  $g$ ,  $n$  和  $x$  都可以是非整数，这时采用线性插值进行计算。

⑧ @ppl( $a, x$ )

Poisson 分布的线性损失函数，即返回  $\max(0, z-x)$  的期望值，其中随机变量  $z$  服从均值为  $a$  的 Poisson 分布。

⑨ @pps( $a, x$ )

均值为  $a$  的 Poisson 分布的累积分布函数。当  $x$  不是整数时，采用线性插值进行计算。

⑩ @psl( $x$ )

单位正态线性损失函数，即返回  $\max(0, z-x)$  的期望值，其中随机变量  $z$  服从标准正态分布。

⑪ @psn( $x$ )

标准正态分布的累积分布函数。

⑫ @ptd( $n, x$ )

自由度为  $n$  的  $t$  分布的累积分布函数。

## ⑬ @qrand(seed)

产生服从(0,1)区间的拟随机数。@qrand 只允许在模型的数据部分使用，它将用拟随机数填满集属性。通常，声明一个  $m \times n$  的二维表， $m$  表示运行实验的次数， $n$  表示每次实验所需的随机数的个数。在行内，随机数是独立分布的；在行间，随机数是非常均匀的。这些随机数是用“分层取样”的方法产生的。

## ⑭ @rand(seed)

返回 0 和 1 间的伪随机数，依赖于指定的种子。典型用法是  $U(I+1)=\text{@rand}[U(I)]$ 。注意如果 seed 不变，那么产生的随机数也不变。

## (4) 变量界定函数

变量界定函数实现对变量取值范围的附加限制，共 4 种：

@bin( $x$ )限制  $x$  为 0 或 1

@bnd( $L, x, U$ )限制  $L \leq x \leq U$

@free( $x$ )取消对变量  $x$  的默认下界为 0 的限制，即  $x$  可以取任意实数

@gin( $x$ )限制  $x$  为整数

在默认情况下，Lingo 规定变量是非负的，也就是说下界为 0，上界为  $+\infty$ 。@free 取消





了默认的下界为 0 的限制, 使变量也可以取负值。@bnd 用于设定一个变量的上下界, 它也可以取消默认下界为 0 的约束。

#### (5) 集操作函数

Lingo 提供了几个函数帮助处理集。

##### ① @in(set\_name, primitive\_index\_1[, primitive\_index\_2, ...])

如果元素在指定集中, 返回 1; 否则返回 0。

##### ② @index([set\_name,] primitive\_set\_element)

该函数返回在集 set\_name 中原始集成员 primitive\_set\_element 的索引。如果 set\_name 被忽略, 那么 Lingo 将返回与 primitive\_set\_element 匹配的的第一个原始集成员的索引。如果找不到, 则产生一个错误。

##### ③ @wrap(index, limit)

该函数返回  $j = \text{index} - k * \text{limit}$ , 其中  $k$  是一个整数, 取适当值保证  $j$  落在区间  $[1, \text{limit}]$  内。该函数相当于 index 模 limit。该函数在循环、多阶段计划编制中特别有用。

##### ④ @size(set\_name)

该函数返回集 set\_name 的成员个数。在模型中明确给出集大小时最好使用该函数。它的使用使模型更加数据中立, 集大小改变时也更易维护。

#### (6) 输入和输出函数

输入和输出函数可以把模型和外部数据比如文本文件、数据库和电子表格等连接起来。

##### ① @file 函数

该函数用从外部文件中输入数据, 可以放在模型中任何地方。该函数的语法格式为 @file('filename')。这里 filename 是文件名, 可以采用相对路径和绝对路径两种表示方式。@file 函数对同一文件的两种表示方式的处理和对两个不同的文件处理是一样的, 这一点必须注意。

把记录结束标记 (~) 之间的数据文件部分称为记录。如果数据文件中没有记录结束标记, 那么整个文件被看作单个记录。注意到除了记录结束标记外, 模型的文本和数据同它们直接放在模型里是一样的。

我们来看一下在数据文件中的记录结束标记连同模型中 @file 函数调用是如何工作的。当在模型中第一次调用 @file 函数时, Lingo 打开数据文件, 然后读取第一个记录; 第二次调用 @file 函数时, Lingo 读取第二个记录, 等等。文件的最后一条记录可以没有记录结束标记, 当遇到文件结束标记时, Lingo 会读取最后一条记录, 然后关闭文件。如果最后一条记录也有记录结束标记, 那么直到 Lingo 求解完当前模型后才关闭该文件。如果多个文件保持打开状态, 可能会导致一些问题, 因为这会使同时打开的文件总数超过允许同时打开文件的上限 16。

当使用 @file 函数时, 可把记录的内容 (除了一些记录结束标记外) 看作是替代模型中 @file('filename') 位置的文本。这也就是说, 一条记录可以是声明的一部分, 整个声明, 或一系列声明。在数据文件中注释被忽略。注意在 Lingo 中不允许嵌套调用 @file 函数。

## ② @text 函数

该函数被用在数据部分用来把解输出至文本文件中。它可以输出集成员和集属性值。其语法为

@text(['filename'])

这里 filename 是文件名，可以采用相对路径和绝对路径两种表示方式。如果忽略 filename，那么数据就被输出到标准输出设备（大多数情形都是屏幕）。@text 函数仅能出现在模型数据部分的一条语句的左边，右边是集名（用来输出该集的所有成员名）或集属性名（用来输出该集属性的值）。

我们把用接口函数产生输出的数据声明称为输出操作。输出操作仅当求解器求解完模型后才执行，执行次序取决于其在模型中出现的先后。

## ③ @ole 函数

@OLE 是从 Excel 中引入或输出数据的接口函数，它是基于传输的 OLE 技术。OLE 传输直接在内存中传输数据，并不借助于中间文件。当使用 @OLE 时，Lingo 先装载 Excel，再通知 Excel 装载指定的电子数据表，最后从电子数据表中获得 Ranges。为了使用 OLE 函数，必须有 Excel 5 及其以上版本。OLE 函数可在数据部分和初始部分引入数据。

@OLE 可以同时读集成员和集属性，集成员最好用文本格式，集属性最好用数值格式。原始集每个集成员需要一个单元(cell)，而对于  $n$  元的派生集每个集成员需要  $n$  个单元，这里第一行的  $n$  个单元对应派生集的第一个集成员，第二行的  $n$  个单元对应派生集的第二个集成员，依此类推。

@OLE 只能读一维或二维的 Ranges（在单个的 Excel 工作表中），但不能读间断的或三维的 Ranges。Ranges 是自左而右、自上而下来读。

## ④ @ranged(variable\_or\_row\_name)

为了保持最优基不变，变量的费用系数或约束行的右端项允许减少的量。

## ⑤ @rangeu(variable\_or\_row\_name)

为了保持最优基不变，变量的费用系数或约束行的右端项允许增加的量。

## ⑥ @status()

返回 Lingo 求解模型结束后的状态：

0 GlobalOptimum（全局最优）

1 Infeasible（不可行）

2 Unbounded（无界）

3 Undetermined（不确定）

4 Feasible（可行）

5 InfeasibleorUnbounded（通常需要关闭“预处理”选项后重新求解模型，以确定模型究竟是不可行还是无界）

6 LocalOptimum（局部最优）

7 LocallyInfeasible（局部不可行，尽管可行解可能存在，但是 Lingo 并没有找到一个）





8 Cutoff (目标函数的截断值被达到)

9 NumericError (求解器因在某约束中遇到无定义的算术运算而停止)

通常, 如果返回值不是 0、4 或 6 时, 那么解将不可信, 几乎不能用。该函数仅被用在模型的数据部分来输出数据。

⑦ @dual

@dual(variable\_or\_row\_name)返回变量的判别数(检验数)或约束行的对偶(影子)价格(dualprices)。

(7) 辅助函数

① @if(logical\_condition,true\_result,false\_result)

@if 函数将评价一个逻辑表达式 logical\_condition, 如果为真, 返回 true\_result, 否则返回 false\_result。

② @warn('text',logical\_condition)

如果逻辑条件 logical\_condition 为真, 则产生一个内容为'text'的信息框。

### 3. Lingo 注意事项

(1) Lingo 中模型以“MODEL:”开始, 以“END”结束, 对于简单的模型, 这两个语句都可以省略;

(2) Lingo 中每行后面均增加了一个分号“;”;

(3) 所有符号都需在英文状态下输入;

(4) min=函数, max=函数, 表示求函数的最小, 最大值;

(5) Lingo 中变量不区分大小写, 变量名可以超过 8 个, 不能超过 32 个, 需以字母开头;

(6) 用 Lingo 解优化模型时已假定所有变量非负, 如果想解除这个限制可以用函数 @free(x), 这样 x 可以取到任意实数;

(7) 变量可以放在约束条件右端, 同时数字也可以放在约束条件左边;

(8) Lingo 模型语句由一系列语句组成, 每一个语句都必须以“;”结尾;

(9) Lingo 中以“!”开始的是说明语句, 说明语句也以“;”结束。

## 3.2.2 Lingo实例

### 1. 简单线性规划求解

目标函数最大值

$$\max \quad z = 4x_1 + 3x_2$$

约束条件

$$\begin{cases} 2x_1 + x_2 \leq 10 \\ x_1 + x_2 \leq 8 \\ x_2 \leq 7 \\ x_1, x_2 \geq 0 \end{cases}$$

Lingo 程序

```
model:
    max=4*x1+3*x2;
    2*x1+x2<10;
    x1+x2<8;
    x2<7;
end
```

注：Lingo 中“<”代表“<=”，“>”代表“>=”，

Lingo 中默认的变量都是大于等于 0 的，不用显式给出。

求解结果： $z=26$ ， $x_1=2$ ， $x_2=6$

## 2. 整数规划求解

$$\text{Max } z = 40x_1 + 90x_2$$

$$\begin{cases} 9x_1 + 7x_2 \leq 56 \\ 7x_1 + 20x_2 \leq 70 \\ x_1, x_2 \geq 0 \end{cases}$$

Lingo 程序

```
model:
    max=40*x1+90*x2;
    9*x1+7*x2<56;
    7*x1+20*x2<70;
    @gin(x1);@gin(x2);
end
```

求解结果： $z=340$ ， $x_1=4$ ， $x_2=2$

## 3. 0-1 规划求解

$$\text{Max } f = x_1^2 + 0.4x_2 + 0.8x_3 + 1.5x_4$$

$$3x_1 + 2x_2 + 6x_3 + 10x_4 \leq 10$$

$$x_1, x_2, x_3, x_4 = 0 \text{ 或 } 1$$

Lingo 程序







```

model:
    max=x1^2+0.4*x2+0.8*x3+1.5*x4;
    3*x1+2*x2+6*x3+10*x4<10;
    @bin(x1);@bin(x2);
    @bin(x3);@bin(x4);
end

```

求解结果:  $f=1.8$ ,  $x_1=1$ ,  $x_2=0$ ,  $x_3=1$ ,  $x_4=0$

#### 4. 非线性规划求解

$$\min \quad z = |x_1| - 2|x_2| - 3|x_3| + 4|x_4|$$

$$\begin{cases} x_1 - x_2 - x_3 + x_4 = 0 \\ x_1 - x_2 + x_3 - 3x_4 = 1 \\ x_1 - x_2 - 2x_3 + 3x_4 = -\frac{1}{2} \end{cases}$$

Lingo 程序

```

model:
    max=@abs(x1)-2*@abs(x2)-3*@abs(x3)+4*@abs(x4);
    x1-x2-x3+x4=0;
    x1-x2+x3-3*x4=1;
    x1-x2-2*x3+3*x4=-1/2;
    @free(x1);@free(x2);
    @free(x3);@free(x4);
end

```

求解结果:  $z=1.25$ ,  $x_1=0.25$ ,  $x_2=0$ ,  $x_3=0$ ,  $x_4=-0.25$

#### 5. 指派问题

某两个煤厂  $A_1$ ,  $A_2$  每月进煤数量分别为 60t 和 100t, 联合供应 3 个居民区  $B_1$ ,  $B_2$ ,  $B_3$ . 3 个居民区每月对煤的需求量依次分别为 50t, 70t, 40t, 煤厂  $A_1$  离 3 个居民区  $B_1$ ,  $B_2$ ,  $B_3$  的距离依次分别为 10km, 5km, 6km, 煤厂  $A_2$  离 3 个居民区  $B_1$ ,  $B_2$ ,  $B_3$  的距离分别为 4km, 8km, 12km. 问如何分配供煤量使得运输量 (即 t·km) 达到最小?

设: 煤厂进煤量  $s_i$ , 居民区需求量为  $d_i$ , 煤厂  $i$  距居民区  $j$  的距离为  $L_{ij}$ , 煤厂  $i$  供给居民区  $j$  的煤量为  $g_{ij}$ .

那么可以列出如下优化方程式

$$\min = \sum_{j=1}^3 \sum_{i=1}^2 g_{ij} \times L_{ij}$$

$$\sum_{i=1}^2 g_{ij} = d_j \quad (j=1,2,3)$$

$$\sum_{j=1}^3 g_{ij} \leq s_i \quad (i=1,2)$$

```

model:
sets:
supply/1,2/:s;
demand/1,2,3/:d;
link(supply,demand):road,sd;
endsets
data:
road=10,5,6,
    4,8,12;
d=50,70,40;
s=60,100;
enddata
[obj]min=@sum(link(i,j):road(i,j)*sd(i,j));
@for(demand(i):@sum(supply(j):sd(j,i)=d(i));
@for(supply(i):@sum(demand(j):sd(i,j)<s(i));
end

```

结果:

```

MIN=940
SD(1,1)=0,SD(1,2)=20,
SD(1,3)=40,SD(2,1)=50,
SD(2,2)=50,SD(2,3)=0

```

## 6. 图论问题

求出图 3-6 所示的最小费用和最大流量，以及在最小费用下的最大流量和最大流量下的最小费用。其中  $(x, y)$  中  $x$  表示容量， $y$  表示费用。

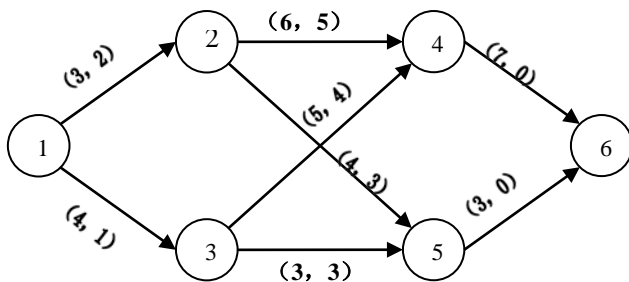


图 3-6

(1) 求最小费用。

解法一：稀疏矩阵 0-1 规划法。

假设图中有  $n$  个原点，现要求从定点 1 到  $n$  的最短路线。设决策变量为  $f_{ij}$ ，当  $f_{ij} = 1$ ，





说明弧 $(i,j)$ 位于定点 1 至定点  $n$  的路线上；否则  $f_{ij} = 0$ ，其数学规划表达式为

$$\min \sum_{i=1}^n \sum_{j=1}^n w_{ij} f_{ij}$$

约束条件，源点只有一条路线出去，终点只有一条路线进来，其余各点进去的和出去的路线相等，表达式如下，

$$\sum_{j=1}^n f_{ij} - \sum_{j=1}^n f_{ji} = \begin{cases} 1 & i=1, \\ -1 & i=n, \\ 0 & i \neq 1, n \end{cases}$$

Lingo 程序

```
model:
sets:
node/1..6/;
road(node,node)/1,2,1,3,2,4,2,5,3,4,3,5,4,6,5,6/:w,f;
endsets
data:
w=2,1,5,3,4,3,0,0;
enddata
n=@size(node);
[obj]min=@sum(road(i,j):w(i,j)*f(i,j));
@for(node(i)|i#ne#1#and#i#ne#n:@sum(road(i,j):f(i,j))=@sum(road(j,i):f(j,i)));
@sum(road(i,j)|i#eq#1:f(i,j))=1;
!下面这个条件可以省略，这个条件包含在上面的条件中了；
!因为如果满足上面所以的条件指向终点的路线只有且仅有一条；
@sum(road(j,i)|i#eq#n:f(j,i))=1;
end
```

结果：

```
Min=4, f(1,3)=1, f(3,5)=1, f(5,6)=1
```

解法二：求源点到任意点的最小费用，动态规划法。

求  $1 \rightarrow 6$  的最小费用，只要求  $1 \rightarrow 4 + 4 \rightarrow 6$  和  $1 \rightarrow 5 + 5 \rightarrow 6$  中的最小费用，以同样的方法向上推，求  $1 \rightarrow 4$  的最小费用只要求出  $1 \rightarrow 2 + 2 \rightarrow 4$  和  $1 \rightarrow 3 + 3 \rightarrow 4$  中的最小费用即可。可以归纳出如下的表达式：

$$L(1) = 0$$

$$L(i) = \min_{j \neq i} (L(j) + c(j,i)), \quad i \neq 1$$

Lingo 程序

```

model:
sets:
node/1..6/:L;
road(node,node)/1,2,1,3,2,4,2,5,3,4,3,5,4,6,5,6/:c;
endsets
data:
c=2,1,5,3,4,3,0,0;
enddata
L(1)=0;
!求一点到任意点的最小费用;
@for(node(i)|i#gt#1:L(i)=@min(road(j,i)|j#ne#i:(L(j)+c(j,i))));
end

```

结果:

```
L(1)=0,L(2)=2,L(3)=1,L(4)=5,L(5)=4,L(6)=4
```

解法三：邻接矩阵法。

如果  $(v_i, v_j) \in E$ ，则称  $v_j$  与  $v_i$  邻接，具有  $n$  个顶点的图的邻接矩阵是一个  $n \times n$  阶矩阵  $A = (a_{ij})_{n \times n}$ ，其分量为

$$a_{ij} = \begin{cases} 1, & (v_i, v_j) \in E \\ 0, & \text{其他} \end{cases}$$

$n$  个顶点的赋权图的赋权矩阵是一个  $n \times n$  阶矩阵  $W = (w_{ij})_{n \times n}$ ，其分量为

$$w_{ij} = \begin{cases} w(v_i, v_j), & (v_i, v_j) \in E \\ \infty, & \text{其他} \end{cases}$$

只需将动态规划的条件改一下即可。

$$L(1) = 0$$

$$L(i) = \min_{j \neq i} [L(j) + a(i, j) \times w(i, j)], \quad i \neq 1, \quad a(i, j) \neq 0$$

Lingo 程序

```

model:
sets:
node/1..6/:L;
road(node,node):a,w;
endsets
data:
a=0,1,1,0,0,0,
0,0,0,1,1,0,

```





```

0,0,0,1,1,0,
0,0,0,0,0,1,
0,0,0,0,0,1,
0,0,0,0,0,0;
w=9,2,1,9,9,9,
9,9,9,5,3,9,
9,9,9,4,3,9,
9,9,9,9,9,0,
9,9,9,9,9,0,
9,9,9,9,9,9;
enddata
L(1)=0;
!求一点到任意点的最小费用;
@for(node(i)|i#gt#1:L(i)=@min(road(j,i)|
j#ne#i#and#a(j,i)#ne#0:(L(j)+w(j,i))));
end

```

结果:

```
L(1)=0,L(2)=2,L(3)=1,L(4)=5,L(5)=4,L(6)=4
```

(2) 求最大流量:

$$\max v_f \quad v_f = \sum_{j=2}^n f_{1j}$$

$$\sum_{j=1}^n f_{ij} - \sum_{j=1}^n f_{ji} = \begin{cases} v_f & i=1 \\ -v_f & i=n \\ 0 & i \neq 1, n \end{cases}$$

同样也可以用三种方法解, 这里只给出邻接矩阵的解法, 因为邻接矩阵最容易扩展到多个点, 且邻接矩阵用其他的软件非常容易得到。

```

model:
sets:
node/1..6/;
road(node,node):w,a,f;
endsets
data:
a=0,1,1,0,0,0,
0,0,0,1,1,0,
0,0,0,1,1,0,
0,0,0,0,0,1,
0,0,0,0,0,1,
0,0,0,0,0,0;
w=0,1,4,0,0,0,
0,0,0,6,4,0,
0,0,0,5,3,0,

```

```

0,0,0,0,0,7,
0,0,0,0,0,3,
0,0,0,0,0,0;
enddata
max=vf;
@sum(road(i,j)|i#eq#1:f(i,j))=vf;
!用下面的表达也可以;
!@sum(node(i):f(1,i))=vf;
@for(node(i)|i#gt#1#and#i#ne#@size(node):@sum(node(j):f(i,j)*a(i,j))=@sum(node(j):f(j,i)*a(j,i)));
@for(road(i,j):f(i,j)<w(i,j));
!用下面的表达也可以;
!@for(road:@bnd(0,f,w));
end

```

结果:

```
Vf=5,F(1,2)=1,F(1,3)=4,F(2,5)=1,F(3,4)=4,F(4,6)=4,F(5,6)=1
```

### (3) 最大流量下的最小费用

用上面的方法得到的最大流量的走法只是其中的一种，而不是所有的走法，所以需要找出最优解，其中最小费用或者最短路径是最常见的两类。

这里求最大流量下的最小费用，先要求出最大流量，然后流量就是已知条件，再求出最小费用就可以了。最大流量用前面的方法已经求出来了，约束条件和上面的一样，这里用  $f_{ij}$  表示现流量， $x_{ij}$  表示最大流量，即容量。

目标函数

$$\min \sum_{i=1}^n \sum_{j=1}^n w_{ij} \operatorname{sgn}(f_{ij})$$

约束条件，源点流出去的流量是最大流量，终点流进的也是最大流量，其余各点进去的和出去的路线相等，表达式如下，

$$\sum_{j=1}^n f_{ij} - \sum_{j=1}^n f_{ji} = \begin{cases} vf & i=1, \\ -vf & i=n, \\ 0 & i \neq 1, n \end{cases}$$

对应的流量要小于容量

$$f_{ij} \leq x_{ij}$$

这里可以由上一问求出  $vf=5$

Lingo 程序如下:

```

model:
sets:

```







```

node/1..6/;
road(node,node)/1,2,1,3,2,4,2,5,3,4,3,5,4,6,5,6/:w,x,f;
endsets
data:
w=2,1,5,3,4,3,0,0;
x=3,4,6,4,5,3,7,3;
enddata
n=@size(node);
[obj]min=@sum(road(i,j):w(i,j)*f(i,j));
@for(node(i)|i#ne#1#and#i#ne#n:@sum(road(i,j):f(i,j))=@sum(road(j,i):f(
j,i)));
@sum(road(i,j)|i#eq#1:f(i,j))=5;
@sum(road(j,i)|i#eq#n:f(j,i))=5;
@for(road(i,j):f(i,j)<x(i,j));
End

```

结果:

```

Min=23
F(1,2)=1,F(1,3)=4,F(2,5)=1,F(3,4)=2
F(3,5)=2,F(4,6)=2,F(5,6)=3

```

(4) 最小费用下的最大流量, 标志  $i$  为流量, 标志  $j$  为费用, 表示  $i$  到  $j$  的现流量。列出数学规划表达式如下:

目标函数,  $\max v_f \quad v_f = \sum_{j=2}^n s_{1j}$

$$\sum_{j=1}^n s_{ij} - \sum_{j=1}^n s_{ji} = \begin{cases} v_f & i=1, \\ -v_f & i=n, \\ 0 & i \neq 1, n \end{cases}$$

这里最小费用等于 4, 有如下约束条件:

$\sum_{i=1}^6 \text{sgn}(s_{ij}) y_{ij} = 4$   $\text{sgn}$  符号函数, 所有经过的路径的费用为 4。

$s_{ij} \leq x_{ij}$  所有的现流量小于容量。

Lingo 程序:

```

model:
sets:
node/1..6/;
road(node,node):x,y,s;
endsets
data:
x=0,1,4,0,0,0,
0,0,0,6,4,0,
0,0,0,5,3,0,
0,0,0,0,0,7,

```

```

0,0,0,0,0,3,
0,0,0,0,0,0;
y=9,2,1,9,9,9,
9,9,9,5,3,9,
9,9,9,4,3,9,
9,9,9,9,9,0,
9,9,9,9,9,0,
9,9,9,9,9,9;
enddata
!max=vf;
!@sum(road(i,j)|i#eq#1: s(i,j))=vf;
!用下面的表达也可以;
n=@size(node);
MAX=@sum(node(i):s(i,n));
@for(node(i)|i#gt#1#and#i#ne#n:@sum(node(j):s(i,j))=@sum(node(j):s(j,i)
));
@sum(road(i,j):@sign(s(i,j))*y(i,j))=4;
@for(road(i,j):s(i,j)<x(i,j));
@for(road:@gin(s));
!用下面的表达也可以;
!@for(road:@bnd(0,s,x));
end

```

结果:

```

MAX=1
S(1,3)=1,S(3,5)=1,S(5,6)=1

```





## 第4章 常用模型与算法

### 4.1 图论

图论中的基本概念是顶点。在实际问题中,许多对象都可以当作顶点,如公交网络中的各个站点可以是顶点,参加比赛的各个队可以看作是顶点,环球旅游时经过的各个城市可以看作是顶点。

顶点组成的集合称为顶点集,记作 $V$ ;顶点之间的连接称为边,边组成的集合称为边集,记作 $E$ 。这样一个图就由 $(V, E)$ 组成,记作 $G = (V, E)$ 。通常 $G$ 的顶点集用 $V(G)$ 表示, $G$ 的边集用 $E(G)$ 表示。如果图中的顶点和边被赋予具体的含义和权,这样的图就称为网络;这时该网络可表示为 $N = (V, E, W)$ , $W$ 由任意两点之间的权值构成权重矩阵。如在研究环球旅游时,各城市构成顶点,城市之间的连接构成边,城市之间的距离就是这两点间边的权重。

如果边的连接是有方向的,称为有向图;如果边之间的连接是无向的,则称为无向图。对一个实际问题,到底应该考虑为无向图还是有向图,需要根据实际问题与背景来区分。如考虑环球旅游问题,各城市之间的连接可当作无向图来研究。研究参赛队伍的排名时,两队之间的胜败是需要区分的,这时要考虑为有向图。

许多实际问题都可以抽象为图论模型。在数模竞赛中,不少问题最终也可以利用图论中的知识求解,如全国数模竞赛中1993B的足球队排名问题,1994B的锁具开箱问题,1998B的洪灾巡视问题,2011B的交巡警平台问题,电工杯数模竞赛中2005B的比赛项目排序问题。这些赛题中部分或全部使用到了图论的知识。

本章分为五个部分进行讲解,每个部分尽量从实用出发,讲解少量的知识,然后利用相关知识求解实际问题,并结合赛题进行讲解。在求解时给出相应的程序,便于读者上手操作。

#### 4.1.1 图论中TSP问题及Lingo求解技巧

巡回旅行商问题(Traveling Salesman Problem, TSP),也称为货郎担问题,最早可以追溯到1759年Euler提出的骑士旅行问题。1948年,由美国兰德公司推动,TSP成为近代组合优化领域的一个典型难题,它已经被证明属于NP难题。

图论的方法描述 TSP 问题如下:给出一个图 $G = (V, E)$ ,每条边 $e \in E$ 上有非负权值 $w(e)$ ,寻找 $G$ 的Hamilton圈 $C$ ,使得 $C$ 的总权值 $W(C) = \sum w(e)$ 最小,其中 $e \in E(C)$ 。





几十年来,出现了很多近似优化算法,如近邻法、贪心算法、最近插入法、最远插入法、模拟退火算法以及遗传算法。这里我们重点介绍利用 Lingo 软件进行求解的方法。

### 1. 产品推销问题

设有一个售货员从 10 个城市中的某一个城市出发,去其他 9 个城市推销产品。10 个城市相互间距离如表 4-1 所示。要求售货员每个城市到达一次,走完一轮后,回到原出发城市。问他应如何选择旅行路线,使总路程最短。

表 4-1

城市 距离 城市	1	2	3	4	5	6	7	8	9	10
1	0	7	4	5	8	6	12	13	11	18
2	7	0	3	10	9	14	5	14	17	17
3	4	3	0	5	9	10	21	8	27	12
4	5	10	5	0	14	9	10	9	23	16
5	8	9	9	14	0	7	8	7	20	19
6	6	14	10	9	7	0	13	5	25	13
7	12	5	21	10	8	13	0	23	21	18
8	13	14	8	9	7	5	23	0	18	12
9	11	17	27	23	20	25	21	18	0	16
10	18	17	12	16	19	13	18	12	16	0

我们采用线性规划的方法求解该问题,设城市之间距离用矩阵  $d$  来表示,  $d_{ij}$  表示城市  $i$  与城市  $j$  之间的距离。设 0-1 矩阵  $X$  用来表示经过的各城市之间的路线。设

$$x_{ij} = \begin{cases} 0 & \text{若城市 } i \text{ 不到城市 } j \\ 1 & \text{若城市 } i \text{ 到城市 } j, \text{ 且 } i \text{ 在 } j \text{ 前} \end{cases} \quad (0.1)$$

考虑每个城市后只有一个城市,则

$$\sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1, \quad i=1, \dots, n \quad (0.2)$$

考虑每个城市前只有一个城市,则

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1, \quad j=1, \dots, n \quad (0.3)$$

仅仅以上约束条件不能避免在一次遍历中产生多于一个互不连通回路。为此我们引入额外变量  $u_i$  ( $i=1, \dots, n$ ), 附加以下充分约束条件:

$$u_i - u_j + nx_{ij} \leq n-1, \quad 1 < i \neq j \leq n$$

该约束的解释为  $i$  与  $j$  不会构成回路, 若构成回路,  $x_{ij} = 1$ ,  $x_{ji} = 1$ , 则  $u_i - u_j \leq -1$  且



$u_j - u_i \leq -1$ , 从而  $0 \leq -2$ , 导致矛盾。如  $i, j$  与  $k$  不会构成回路, 若构成回路,  $x_{ij} = 1, x_{jk} = 1, x_{ki} = 1$ , 则  $u_i - u_j \leq -1, u_j - u_k \leq -1, u_k - u_i \leq -1$ , 从而  $0 \leq -3$ , 导致矛盾。其他情况依此类推。

综合上述三个约束条件, 我们得到如下优化模型:

$$\min Z = \sum_{i=1}^n \sum_{j=1}^n d_{ij} x_{ij}$$

$$s.t. \begin{cases} \sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1, & j = 1, \dots, n \\ \sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1, & i = 1, \dots, n \\ u_i - u_j + nx_{ij} \leq n-1, & 1 < i \neq j \leq n \\ x_{ij} = 0 \text{ 或 } 1, & i, j = 1, \dots, n \\ u_i \text{ 为实数}, & i = 1, \dots, n \end{cases}$$

上述优化模型的 Lingo 程序为

```
!TSPquestion;
MODEL:
SETS:
    city/1..10/: u;
    link(city,city): d,x;
ENDSETS
DATA:
d=
0   7   4   5   8   6   12  13  11  18
7   0   3  10   9  14   5  14  17  17
4   3   0   5   9  10  21   8  27  12
5  10   5   0  14   9  10   9  23  16
8   9   9  14   0   7   8   7  20  19
6  14  10   9   7   0  13   5  25  13
12  5  21  10   8  13   0  23  21  18
13 14   8   9   7   5  23   0  18  12
11 17  27  23  20  25  21  18   0  16
18 17  12  16  19  13  18  12  16   0;
ENDDATA
MIN=@SUM(link: d*x);
!城市 j 前有一个城市相连;
@for(city(j): @sum(city(i) | j#ne#i: x(i,j))=1);
!城市 i 后有一个城市相连;
@for(city(i): @sum(city(j) | j#ne#i: x(i,j))=1);
@for(link(i,j) | i#ne#j#and#i#gt#1: u(i)-u(j)+10*x(i,j)<=9);
@FOR(link: @BIN(x));
End
```







求解得到该售货员的行程路线如下： $X(3,2)=1, X(4,1)=1, X(4,3)=1, X(6,5)=1, X(7,2)=1, X(7,5)=1, X(8,6)=1, X(9,1)=1, X(10,8)=1, X(10,9)=1$ ，其他全为 0。对应的最短路线为 1—4—3—2—7—5—6—8—10—9—1，最短距离为 77 千米。

## 2. 比赛项目排序问题

全民健身计划是 1995 年在国务院领导下，由国家体委会同有关部门、各群众组织和社会团体共同推行的一项依托社会、全民参与的体育健身计划，是与实现社会主义现代化目标相配套的社会系统工程和跨世纪的发展战略规划。现在，以全民健身为主要内容的群众性体育活动蓬勃开展，举国上下形成了全民健身的热潮，人民群众健康水平不断提高，同时也扩大了竞技体育的社会影响，提高了竞技体育水平。现在各级、各类、各种运动比赛比比皆是，这不但提高了全民的身体素质，而且使一批运动员脱颖而出，成为运动健将，为国家争得了荣誉。

在各种运动比赛中，为了使比赛公平、公正、合理地举行，一个基本要求是：在比赛项目排序过程中，尽可能使每个运动员不连续参加两项比赛，以便运动员恢复体力，发挥正常水平。

(1) 表 4-2 所列是某个小型运动会的比赛报名表。有 14 个比赛项目，40 名运动员参加比赛。表中第 1 行表示 14 个比赛项目，第 1 列表示 40 名运动员，表中“#”号位置表示运动员参加此项比赛。建立此问题的数学模型，并且合理安排比赛项目顺序，使连续参加两项比赛的运动员人次尽可能的少。

(2) 文件“运动员报名表”（见网址：<http://www.mathor.com/thread-11976-1-1.html>）中给出了某个运动比赛的报名情况。共有 61 个比赛项目，1050 人参加比赛。请给出算法及其框图，同时给出合理的比赛项目排序表，使连续参加两项比赛的运动员人次尽可能的少。

(3) 说明上述算法的合理性。

(4) 对问题 2 的比赛排序结果，给出解决“运动员连续参加比赛”问题的建议及方案。

结果，给出解决“运动员连续参加比赛”问题的建议及方案。

表 4-2

项目 运动员	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1		#	#						#				#	
2								#			#	#		
3		#		#						#				
4			#					#				#		
5											#		#	#
6					#	#								
7												#	#	
8										#				#
9		#		#						#	#			
10	#	#		#			#							

(续表)

项目 运动员	1	2	3	4	5	6	7	8	9	10	11	12	13	14
11		#		#									#	#
12								#		#				
13					#					#				#
14			#	#				#						
15			#					#				#		
16									#		#	#		
17						#								#
18							#					#		
19			#							#				
20	#			#										
21									#					#
22		#			#									
23							#					#		
24							#	#					#	#
25	#	#								#				
26					#									#
27						#					#			
28		#						#						
29	#										#	#		
30				#	#									
31						#		#				#		
32							#			#				
33				#		#								
34	#		#										#	#
35					#	#						#		
36				#			#							
37	#								#	#				
38						#		#		#				#
39					#			#	#				#	
40						#	#		#				#	

• 问题 1

若项目  $i$  和项目  $j$  相邻, 可以计算出同时参加这两个项目的人数, 作为  $i$  和  $j$  的距离  $d_{ij}$ 。

则问题转化为求项目 1 到项目 14 的一个排列, 使相邻距离和最小。我们采用 TSP 问题求解。但由于开始项目和结束项目没有连接, 可考虑引入虚拟项目 15, 该虚拟项目与各个项目的距离都为 0。

求解距离矩阵  $D$  时, 首先将上述报名表用矩阵  $A_{40 \times 14}$  表示, 矩阵中的每个元素  $a_{ij}$  定义为

$$a_{ij} = \begin{cases} 1 & \text{第 } i \text{ 个人参加项目 } j \\ 0 & \text{第 } i \text{ 个人不参加项目 } j \end{cases}$$

则



$$d_{ij} = \sum_{k=1}^{40} a_{ki} \cdot a_{kj} \quad i \neq j, \quad i, j = 1, 2, \dots, 14$$

$$d_{ii} = 0 \quad i = 1, 2, \dots, 14$$

$$d_{i,15} = 0, d_{15,i} = 0 \quad i = 1, 2, \dots, 15$$

由于问题 1 中 40 个运动员参加 14 个项目的比赛是 Word 表,可将其拷贝到 Excel 表中,然后将#替换为 1,将空格替换为 0,形成 0-1 表,并拷贝到数据文件 table1.txt(见网址:<http://www.mathor.com/thread-11976-1-1.html>)中。问题 2 中 1050 个运动员参加的 61 个项目比赛的 Access 数据库中的表保存为 Excel 表,然后在表中将#替换为 1,将空格替换为 0,形成 0-1 表,并拷贝到数据文件 table2.txt(见网址:<http://www.mathor.com/thread-11976-1-1.html>)中。MatLab 求解距离矩阵的程序如下(见网址:<http://www.mathor.com/thread-11976-1-1.html>):

```
load table1.txt;
a = table1;
[m,n] = size(a);
d = zeros(n+1,n+1); %定义距离矩阵;
d(1:n,1:n) = a'*a; %计算不同项目之间距离
d = d - diag(diag(d)); %相同项目间的距离设为 0
%输出文件
fid=fopen('d:\lingo12\dat\dis1.txt','w');
for i=1:n+1
    for j=1:n+1
        fprintf(fid,'%1d ',d(i,j));
    end
    fprintf(fid,'\r\n');
end
fclose(fid);
```

输出的距离矩阵  $D$  为 dis1.txt:

```
0 2 1 2 0 0 1 0 1 2 1 1 1 1 0
2 0 1 4 1 0 1 1 1 3 1 0 2 1 0
1 1 0 1 0 0 0 3 1 1 0 2 2 1 0
2 4 1 0 1 1 2 1 0 2 1 0 1 1 0
0 1 0 1 0 2 0 1 1 1 0 1 1 2 0
0 0 0 1 2 0 1 2 1 1 1 2 1 2 0
1 1 0 2 0 1 0 1 1 1 0 2 2 1 0
0 1 3 1 1 2 1 0 1 2 1 4 2 2 0
1 1 1 0 1 1 1 1 0 1 1 1 3 1 0
2 3 1 2 1 1 1 2 1 0 1 0 0 3 0
1 1 0 1 0 1 0 1 1 1 0 3 1 1 0
1 0 2 0 1 2 2 4 1 0 3 0 1 0 0
1 2 2 1 1 1 2 2 3 0 1 1 0 4 0
```

```

1 1 1 1 2 2 1 2 1 3 1 0 4 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

Lingo 程序 B2005.lg4:

```

!第一个问题的求解程序:
!比赛项目排序问题;

model:
sets:
    item / 1.. 15/: u;
    link( item, item):dist,x;
endsets
    n = @size( item);
data:    !距离矩阵;
dist=@file('d:\lingo12\dat\dis1.txt'); !文件路径;
    !输出为 1 的变量;
@text()=@writefor(link(i,j)|x(i,j)#GT#0:' x(' ,i ,',' ,j ,')=' ,x(i,j));
enddata

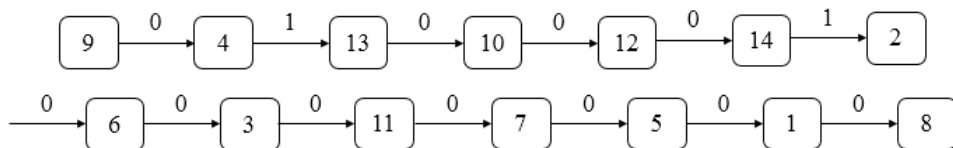
MIN=@SUM(link:dist*x);
@for(item(j):@sum(item(i)|j#ne#i:x(i,j))=1); !点 j 前有一个点相连;
@for(item(i):@sum(item(j)|j#ne#i:x(i,j))=1); !点 i 后前有一个点;
!保证不出现子圈;
@for(link(i,j)|i#NE#j#and#i#gt#1:u(i)-u(j)+n*x(i,j)<=n-1);
@FOR(link:@BIN(x));!定义 x 为 0-1 变量;
end

```

其中数据文件 dis1.txt 在 MatLab 程序 B2005.m 中输出。Lingo12 求解结果为:

目标值  $z=2$ ,  $x(1,8)=1$   $x(2,6)=1$   $x(3,11)=1$   $x(4,13)=1$   $x(5,1)=1$   $x(6,3)=1$   $x(7,5)=1$   
 $x(8,15)=1$   $x(9,4)=1$   $x(10,12)=1$   $x(11,7)=1$   $x(12,14)=1$   $x(13,10)=1$   $x(14,2)=1$   
 $x(15,9)=1$ 。

由于 15 是虚拟项,去掉后对应序列为 9-4-13-10-12-14-2-6-3-11-7-5-1-8-9,则项目排序如下,其中箭头上所示数字为连续参加相邻两项目的运动员数。



即有两名运动员连续参加比赛。

#### • 问题 2

问题 2 解答与问题 1 相同,只是项目变成 61 个,引入虚拟项目后变为 62 个,运动员为 1050 名。模型建立同问题 1。在问题 1 中的 MatLab 程序中只需要将表 table1.txt 改为 table2.txt,输出数据文件将 dis1.txt 改为 dis2.txt 就可以了。





在 Lingo 程序中将项目数由 15 修改为 62, 使用的数据文件由 15 改为 62, 同样可以运行, 只是运行时间较长, 本程序在 Lingo12 中大约运行 6 分钟左右。原始数据文件 table2.txt 和 MatLab 输出的距离矩阵 dis2.txt, 由于数据较大这里不列出, 可参见附录。

问题 2 的 Lingo 程序:

```
!第二个问题的求解的程序:
!比赛项目排序问题;
model:
sets:
    item/1..62/: u;
    link(item,item): dist,x;
endsets
n=@size(item);
data: !距离矩阵;
    dist=@file('d: \lingo12\dat\dis2.txt');!文件路径;
    @text()=@writefor(link(i,j)|x(i,j)#GT#0: 'x(' ,i , ',' ,j ,')=' ,x(i,j));!输出为 1 的变量;
enddata
MIN=@SUM(link: dist*x);
@for(item(j): @sum(item(i)|j#ne#i: x(i,j))=1);!点 j 前有一个点相连;
@for(item(i): @sum(item(j)|j#ne#i: x(i,j))=1);!点 i 后前有一个点;
!保证不出现子圈;
@for(link(i,j)|i#NE#j#and#i#gt#1: u(i)-u(j)+n*x(i,j)<=n-1);
@FOR(link: @BIN(x));!定义 x 为 0-1 变量;
end
```

Lingo12 求解结果为:

目标值  $z=5$ ,  $x(1,19)=1$   $x(2,44)=1$   $x(3,50)=1$   $x(4,25)=1$   $x(5,20)=1$   $x(6,15)=1$   
 $x(7,42)=1$   $x(8,59)=1$   $x(9,35)=1$   $x(10,3)=1$   $x(11,54)=1$   $x(12,21)=1$   $x(13,32)=1$   $x(14,41)=1$   
 $x(15,40)=1$   $x(16,57)=1$   $x(17,22)=1$   $x(18,9)=1$   $x(19,60)=1$   $x(20,6)=1$   $x(21,10)=1$   $x(22,37)=1$   
 $x(23,14)=1$   $x(24,51)=1$   $x(25,13)=1$   $x(26,27)=1$   $x(27,29)=1$   $x(28,17)=1$   $x(29,24)=1$   $x(30,58)=1$   
 $x(31,12)=1$   $x(32,56)=1$   $x(33,47)=1$   $x(34,23)=1$   $x(35,46)=1$   $x(36,45)=1$   $x(37,30)=1$   $x(38,49)=1$   
 $x(39,31)=1$   $x(40,48)=1$   $x(41,1)=1$   $x(42,52)=1$   $x(43,38)=1$   $x(44,4)=1$   $x(45,7)=1$   $x(46,62)=1$   
 $x(47,55)=1$   $x(48,34)=1$   $x(49,26)=1$   $x(50,36)=1$   $x(51,16)=1$   $x(52,18)=1$   $x(53,39)=1$   $x(54,43)=1$   
 $x(55,5)=1$   $x(56,11)=1$   $x(57,53)=1$   $x(58,61)=1$   $x(59,28)=1$   $x(60,8)=1$   $x(61,2)=1$   $x(62,33)=1$ 。

由于 62 是虚拟项, 去掉后对应序列为:

33—47—55—5—20—6—15—40—48—34—23—14—41—1—19—60—8—59—28—  
 17—22—37—30—58—61—2—44—4—25—13—32—56—11—54—43—38—49—  
 26—27—29—24—51—16—57—53—39—31—12—21—10—3—50—36—45—7—  
 42—52—18—9—35—46

可以验证, 其中  $d(14,41)=1$ ,  $d(51,16)=1$ ,  $d(31,12)=1$ ,  $d(10,3)=1$ ,  $d(45,7)=1$ 。其余相邻两个项目比赛没有两名运动员连续参加比赛, 即有 5 名运动员连续参加比赛。另外, 该问



题的解不唯一，在目标值都为5的情况下，比赛项目安排顺序还可以按照下列形式进行：

$x(1,41)=1$   $x(2,61)=1$   $x(3,10)=1$   $x(4,44)=1$   $x(5,55)=1$   $x(6,20)=1$   $x(7,45)=1$   $x(8,60)=1$   
 $x(9,18)=1$   $x(10,21)=1$   $x(11,54)=1$   $x(12,31)=1$   $x(13,32)=1$   $x(14,23)=1$   $x(15,6)=1$   $x(16,57)=1$   
 $x(17,28)=1$   $x(18,52)=1$   $x(19,1)=1$   $x(20,5)=1$   $x(21,12)=1$   $x(22,17)=1$   $x(23,34)=1$   $x(24,51)=1$   
 $x(25,4)=1$   $x(26,27)=1$   $x(27,29)=1$   $x(28,59)=1$   $x(29,24)=1$   $x(30,37)=1$   $x(31,39)=1$   $x(32,56)=1$   
 $x(33,62)=1$   $x(34,48)=1$   $x(35,9)=1$   $x(36,50)=1$   $x(37,22)=1$   $x(38,49)=1$   $x(39,13)=1$   $x(40,15)=1$   
 $x(41,14)=1$   $x(42,7)=1$   $x(43,38)=1$   $x(44,2)=1$   $x(45,36)=1$   $x(46,35)=1$   $x(47,33)=1$   $x(48,40)=1$   
 $x(49,26)=1$   $x(50,3)=1$   $x(51,16)=1$   $x(52,42)=1$   $x(53,25)=1$   $x(54,43)=1$   $x(55,47)=1$   $x(56,11)=1$   
 $x(57,53)=1$   $x(58,30)=1$   $x(59,8)=1$   $x(60,19)=1$   $x(61,58)=1$   $x(62,46)=1$ 。

## 4.1.2 最短路算法及在建模中的应用

图论中的最短路问题（包括无向图和有向图）是一个经常遇到的基本问题，解决该类问题的主要算法有 Dijkstra 算法和 Floyd 算法。其中 Dijkstra 算法是求出指定两点之间的最短路线，算法复杂度为  $O(n^2)$ ；Floyd 算法是求出任意两点之间的最短路，算法复杂度为  $O(n^3)$ 。其中  $n$  为顶点数。

### 1. Dijkstra 算法

假定给出一个网络  $N=(V,E,W)$ ，现在要求出任意点  $i$  到任意点  $j$  之间的最短路，算法描述如下：

(1) 给出初始点集合  $P=\{i\}$ ，剩余点集合  $Q=\{1,2,\dots,n\}-P$ 。初始  $i$  点到各点的直接距离  $U_r=w_{ir}(r=1,2,\dots,n)$ ；

(2) 在  $Q$  中寻找找到  $i$  点距离最小的点  $k$ ，使得  $U_k=\min_{r \in Q}\{U_r\}$ ，并置  $P \cup \{k\} \rightarrow P$ ， $Q - \{k\} \rightarrow Q$ ；

(3) 对  $Q$  中每个  $r$ ，如果  $U_k + w_{kr} < U_r$ ，则  $U_k + w_{kr} \rightarrow U_r$ ，然后返回 (2) 直到找到  $j$  点为止。

注：步骤 (2) 和 (3) 实际上是每当找到一个到  $i$  点距离最小的点  $k$  时，就更新一次从  $i$  点出发，通过集合  $P$  中的点到达  $Q$  中的点的最小距离。

该算法经过  $n-1$  次循环结束。在整个算法过程中，步骤 (2) 最多作  $\frac{1}{2}(n-1)(n-2)$  次比较，步骤 (3) 最多作  $\frac{1}{2}(n-1)(n-2)$  次加法和比较，总的计算量是  $O(n^2)$  阶。因此这个算法是一个有效算法。

### 2. Floyd 算法

(1) 根据已知的部分节点之间的连接信息，建立初始距离矩阵  $B(i,j)$ ，其中没有给出距离的赋予一个充分大数值，以便于更新；( $i,j=1,2,\dots,n$ )；

(2) 进行迭代计算。对任意两点  $(i,j)$ ，若存在  $k$ ，使  $B(i,k)+B(k,j) < B(i,j)$ ，则更





新  $B(i, j) = B(i, k) + B(k, j)$ ;

(3) 直到所有点的距离不再更新停止计算。则得到最短路距离矩阵  $B(i, j)$ ,  $(i, j = 1, 2, \dots, n)$ 。

算法程序为:

```
for k=1:n
for i=1 :n
    for j=1:n
        t=B(i,k)+B(k,j);
        if t<B(i,j) B(i,j)=t; end
    end
end
end
```

### 3. 案例 1 灾情巡视路线问题 (CUMCM1998B)

图 4-1 为某县的乡(镇)、村公路网示意图,公路边的数字为该路段的千米数。今年夏天该县遭受水灾。为考察灾情、组织自救,县领导决定,带领有关部门负责人到全县各乡(镇)、村巡视。巡视路线指从县政府所在地出发,走遍各乡(镇)、村,又回到县政府所在地的路线。

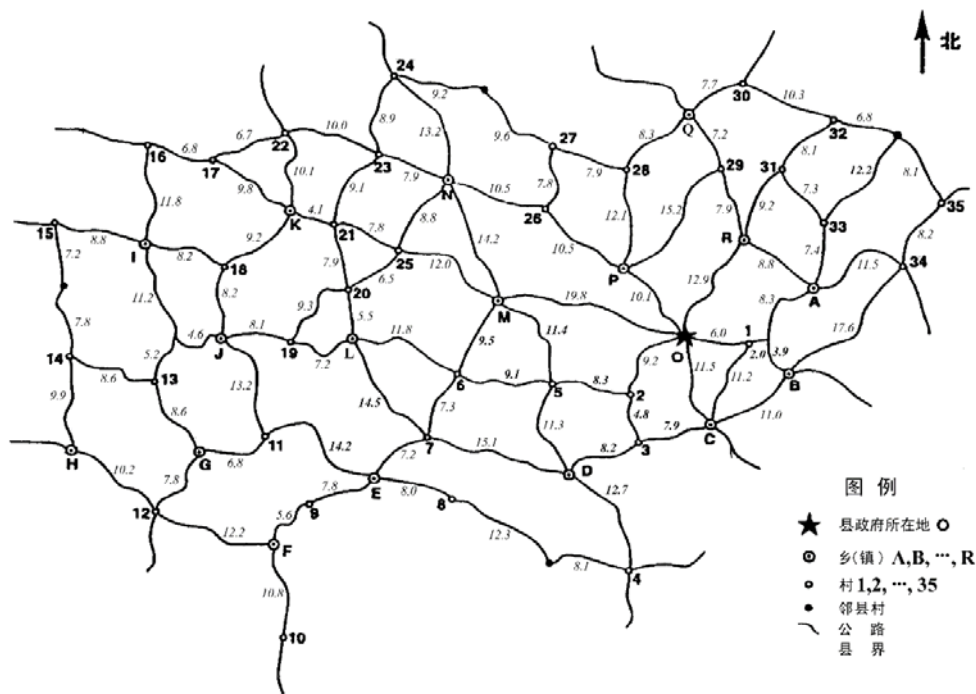


图 4-1



(1) 若分三组(路)巡视,试设计总路程最短且各组尽可能均衡的巡视路线。

(2) 假定巡视人员在各乡(镇)停留时间  $T=2h$ , 在各村停留时间  $t=1h$ , 汽车行驶速度  $V=35km/h$ 。要在  $24h$  内完成巡视,至少应分几组;给出这种分组下你认为最佳的巡视路线。

(3) 在上述关于  $T$ ,  $t$  和  $V$  的假定下,如果巡视人员足够多,完成巡视的最短时间是多少;给出在这种最短时间完成巡视的要求下,你认为最佳的巡视路线。

(4) 若巡视组数已定(如三组),要求尽快完成巡视,讨论  $T$ ,  $t$  和  $V$  改变对最佳巡视路线的影响。

本题给出了某县的公路网络图,要求的是在不同的条件下,灾情巡视的最佳分组方案和路线。将每个乡(镇)或村看作一个图的顶点,各乡镇、村之间的公路看作此图对应顶点间的边,各条公路的长度(或行驶时间)看作对应边上的权,所给公路网就转化为加权网络图,问题就转化图论中一类称之为旅行商的问题,即在给定的加权网络图中寻找从给定点  $O$  出发,行遍所有顶点至少一次再回到点  $O$ ,使得总权(路程或时间)最小。

该题是旅行商问题的延伸——多旅行商问题。所求分组巡视的最佳路线,也就是  $m$  条经过同一点并覆盖所有其他顶点,又使边权之和达到最小的闭链。如第一问是三个旅行商问题,第二问是四个旅行商问题。众所周知,旅行商问题属于 NP 难题,即求解没有多项式时间算法。

显然,该问题更应属于 NP 难题,有鉴于此,一定要针对问题的实际特点寻找简便方法,想找到解决此类问题的一般方法是不现实的,对于规模较大的问题可使用近似算法来求得近似最优解。

在本题中首先用图论中 Floyd 算法求出  $G$  中任意两个顶点间的最短路径,并构造出完全图  $G'=(V,E')$ ,  $\forall(x,y)\in E',\omega(x,y)=\min d_G(x,y)$ 。问题转化为在给定的加权网络图中寻找从给定点  $O$  出发,行遍所有顶点至少一次再回到点  $O$  的线路,使得该线路的总权(路程或时间)最小,即最佳旅行商问题;各问题的详细解答陈述如下。

#### ◆ 问题 1

问题要求:若分为三组巡视,设计总路程最短且各组尽可能均衡的巡视路线。此问题是多个旅行商的最佳旅行商问题。即在加权图  $G$  中求顶点集  $V$  的划分  $V_1,V_2,V_3$ ,将  $G$  分成 3 个生成子图  $G[V_1],G[V_2],G[V_3]$ ,使得

(1) 顶点  $O\in V_i$ ,  $i=1,2,3$ ;

(2)  $\bigcup_{i=1}^3 V_i = V(G)$ ;

(3)  $\frac{\max_{i,j} |\omega(C_i) - \omega(C_j)|}{\max_i \omega(C_i)} \leq \alpha$ , 其中  $C_i$  为  $V_i$  的导出子图  $G[V_i]$  中的最佳旅行售货员回路,  $\omega(C_i)$  为  $C_i$  的权(总路程),  $i,j=1,2,3$ ;

(4)  $\sum_{i=1}^3 \omega(C_i) = \min \cdots$  总路程最小。





我们定义称  $\alpha_0 = \frac{\max_{i,j} |\omega(C_i) - \omega(C_j)|}{\max_i \omega(C_i)}$  为该分组的实际均衡度， $\alpha$  为最大容许均衡度。

显然  $0 \leq \alpha_0 \leq 1$ ， $\alpha_0$  越小，说明分组的均衡性越好。取定一个  $\alpha$  后， $\alpha_0$  与  $\alpha$  满足条件 (3) 的分组是一个均衡分组。条件 (4) 表示总巡视路线最短。此问题包含两方面：①对顶点分组；②在每组中求（单个售货员）最佳旅行商回路。

因单个旅行商的最佳旅行商回路问题也不存在多项式时间内的精确算法。而图中节点数较多（53 个），我们只能去寻求一种较合理的划分准则，对图 4-1 进行粗步划分后，求出各部分的近似最佳旅行商回路的权，再进一步调整，使得各部分满足均衡性条件 (3)。

从 O 点出发去其他点，要使路程较小应尽量走 O 点到该点的最短路。利用 Floyd 算法中同时求出各点从 O 点出发的前点。将每点和它对应的前点连接起来，得到如图 4-2 所示的树状图。该图可以看作从 O 出发由 6 条树干构成的树。

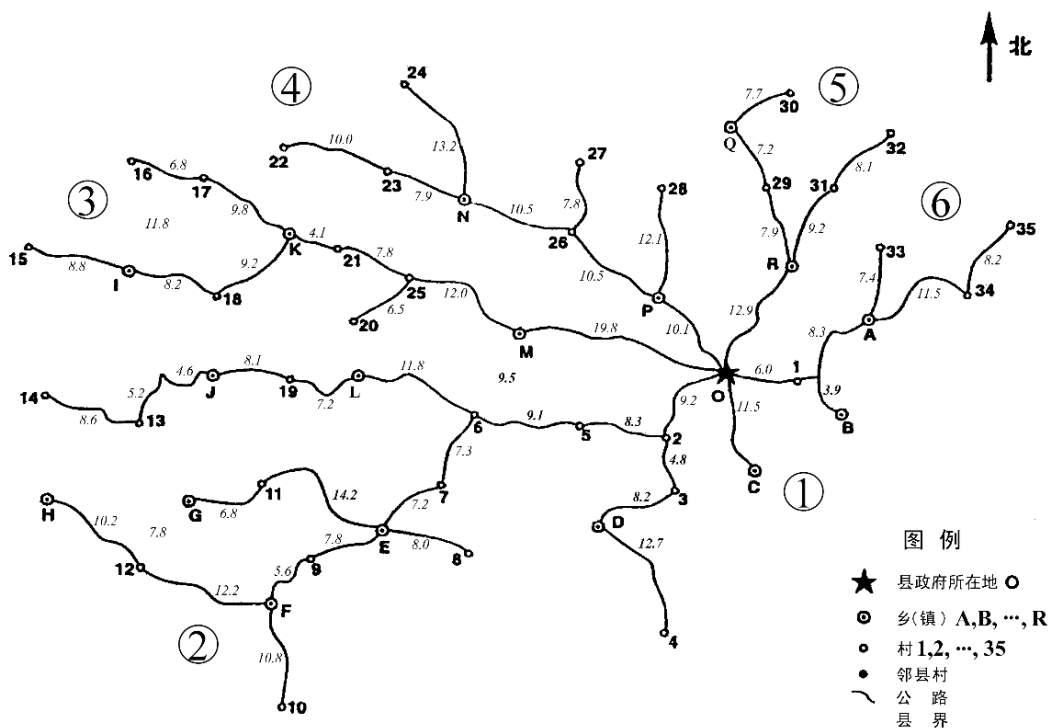


图 4-2

由上述分组准则，我们找到两种分组形式如下。

分组 1: (⑥, ①), (②, ③), (⑤, ④);

分组 2: (①, ②), (③, ④), (⑤, ⑥)。

分组 1 中①和⑥构成的一组点太少，显然这导致与其余两组差异太大，故考虑分组 2 的情况 (①, ②), (③, ④), (⑤, ⑥)。对分组 2 中每组顶点的生成子图如图 4-3 所示，

采用优化算法求出近似最优解及相应的巡视路线，如表 4-3 所示。

表 4-3

小 组	路 线	每组路线总长	路线总长度
I	O—P—28—27—26—N—24—23—22—17—16—I—15—I—18—K—21—20—25—M—O	191.1	558.5
II	O—2—5—6—L—19—J—11—G—13—14—H—12—F—10—F—9—E—7—E—8—4—D—3—C—O	241.9	
III	O—R—29—Q—30—32—31—33—35—34—A—B—1—O	125.5	

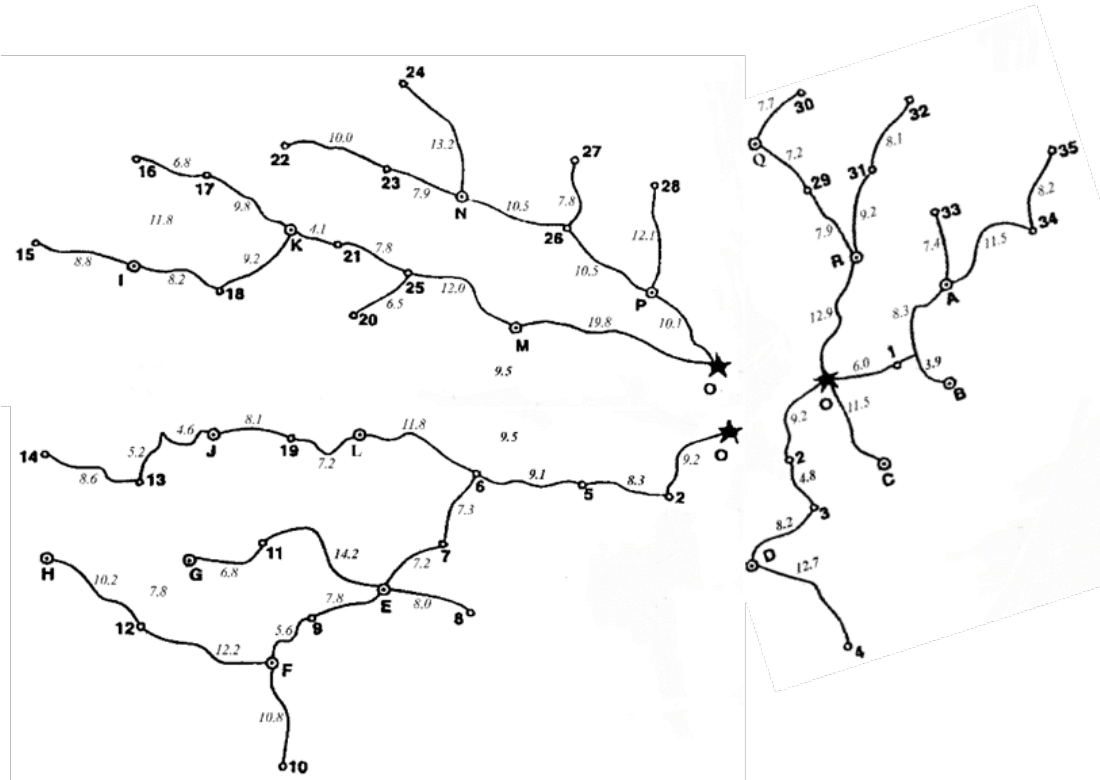


图 4-3

该分组的均衡度

$$\alpha_0 = \frac{\omega(C_1) - \omega(C_2)}{\max_{i=1,2,3} \omega(C_i)} = \frac{241.9 - 125.5}{241.9} = 54.2\%$$

从结果看该分法的均衡性很差。为改善均衡性，将第 II 组中的顶点 C，2，3，D，4 分给第 III 组（顶点 2 为这两组的公共点），重新分组后的近似最优解划分方案见表 4-4 所示。



表 4-4

小 组	路 线	每组路 线总长	路线 总长度
I	O—P—28—27—26—N—24—23—22—17—16—I—15—I—18—K—21—20—25—M—O	191.1	558.5
II	O—2—5—6—7—E—8—E—9—F—10—F—12—H—14—13—G—11—J—19—L—6—5—2—O	216.4	
III	O—E—29—Q—30—32—31—33—35—34—A—1—B—C—3—D—4—D—3—2—O	192.3	

接下来讲述  $n$  个点之间最优巡视线路的模型, 设两个点之间距离用矩阵  $d$  来表示,  $d_{ij}$  表示点  $i$  与点  $j$  之间的距离。设 0-1 矩阵  $x$  用来表示经过的各点之间的路线。设

$$x_{ij} = \begin{cases} 0 & \text{若 } i \text{ 不到 } j \\ 1 & \text{若 } i \text{ 到 } j, \text{ 且 } i \text{ 在 } j \text{ 前} \end{cases}$$

则最优巡视线路的目标函数为  $\min z = \sum_{i,j=1}^n d_{ij} x_{ij}$

约束满足:

考虑每个点后只有一个点, 则

$$\sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1, \quad i = 1, \dots, n$$

考虑每个点前只有一个点, 则

$$\sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1, \quad j = 1, \dots, n$$

但仅以上约束条件不能避免在一次遍历中产生多于一个互不连通回路。为此我们引入额外变量  $u_i$  ( $i = 1, \dots, n$ ), 附加以下充分约束条件:  $u_i - u_j + nx_{ij} \leq n-1, \quad 1 < i \neq j \leq n$ ;

于是我们可以得到如下的模型 (实现程序见 Lingo 程序 B98.lg4)

$$\begin{aligned} \min z &= \sum_{i,j=1}^n d_{ij} x_{ij} \\ \text{s.t.} \quad &\left\{ \begin{array}{l} \sum_{\substack{i=1 \\ i \neq j}}^n x_{ij} = 1, \quad j = 1, \dots, n \\ \sum_{\substack{j=1 \\ j \neq i}}^n x_{ij} = 1, \quad i = 1, \dots, n \\ u_i - u_j + nx_{ij} \leq n-1, \quad 1 < i \neq j \leq n \\ x_{ij} = 0 \text{ 或 } 1, \quad i, j = 1, \dots, n \\ u_i \text{ 为实数}, \quad i = 1, \dots, n \end{array} \right. \end{aligned}$$

利用该模型可以计算出当选定一组点后的最优路线, 附录中有相应程序。

因该分组的均衡度

$$\alpha_0 = \frac{\omega(C_3) - \omega(C_1)}{\max_{i=1,2,3} \omega(C_i)} = \frac{216.4 - 191.1}{216.4} = 11.69\%$$

所以这种分法的均衡性较好，故采用表 4-2 方案。

#### ◆ 问题 2

当巡视人员在各乡（镇）、村的停留时间一定，汽车的行驶速度一定，要在 24h 内完成巡视，至少要分几组及最佳旅行的巡视路线。

由于  $T=2\text{ h}$ ， $t=1\text{ h}$ ， $V=35\text{ km/h}$ ，需访问的乡镇共有 17 个，村共有 35 个。计算出在乡（镇）及村的总停留时间为  $17 \times 2 + 35 = 69\text{ h}$ ，要在 24h 内完成巡回，若分成 3 组，由于每组分担停留时间为  $69/3=23\text{ h}$ ，接近 24h，加上还有一定行走时间，因此 3 组肯定不行，我们考虑分成 4 组。

由于该网络的乡（镇）、村分布较为均匀，故有可能找出停留时间尽量均衡的分组，当分 4 组时各组停留时间大约为  $69/4=17.25\text{ h}$ ，则每组分配在路途上的时间大约为  $24-17.25=6.75\text{ h}$ 。而前面讨论过，分三组时有个总路程 599.8km 的巡视路线，分 4 组时的总路程不会比 599.8km 大太多，不妨以 599.8km 来计算。路上约用  $599.8/35=17\text{ h}$ ，若平均分配给 4 个组，每个组约需  $17/4=4.25\text{ h} < 6.75\text{ h}$ ，故分成 4 组是可能办到的。

现在尝试将顶点分为 4 组，分组的原则：除遵从前面准则 1、2、3 外，还应遵从以下准则：准则 4 尽量使各组的停留时间相等。

用上述原则分为 4 组，同时计算各组的停留时间，然后用算法一算出各组的近似最佳旅行商巡回，得出路线长度及行走时间，这 4 组的近似最优解见表 4-5 所列。

表 4-5

小 组	路 线	路线 总长度	停留 时间	行走 时间	路线 总长度
I	O—2—5—6—7—E—8—E—11—G—12—H—12—F—10—F—9—E—7— —6—5—2—O	225.3	17	6.44	23.44
II	O—R—29—Q—30—Q—28—27—26—N—24—23—22—17—16—17— K—22—23—N—26—P—O	199.2	16	5.69	21.69
III	O—M—25—20—21—K—18—I—15—14—13—J—19—L—6—M—O	166.3	18	4.75	22.75
IV	O—R—A—33—31—32—35—34—B—I—C—3—D—4—D—3—2—O	162.9	18	4.65	22.65

表中，加有底纹的点表示前面已经经过并停留过了，此次只经过不停留；加粗的点表示此点只经过不停留。

该分组实际均衡度  $\alpha_0 = \frac{(22.74 - 21.69)}{22.74} = 7.5\%$ ，可看出，表 4-5 分组的均衡度很好，且完全满足 24 h 完成巡视的要求。

#### ◆ 问题 3

经计算，从 O 出发，最远的点为 H 点，距离为 77.5km，则到该镇所花时间为：

$$\text{TIME} = 77.5 \times 2 / 35 + 2 = 6.43$$



因此当巡视人员足够多时,完成巡视的时间为 6.43 h。

#### 程序 1: MatLab 程序 B98.m

该程序并没有直接计算出最后结果。该程序主要求出了各点的最短路矩阵,各点到县城  $O$  点的前点,便于做出图 4-1 那样的树状图。对问题 2 和问题 3,在于对给定的路线进行总路程计算,相当于最后的验证程序。

```
%98B 灾情巡视问题
n=53; %总共 53 个点
%将 1,2,3,...,35 个村标号不变
%将 A,B,C,...,Q,R 共 18 个镇分别标号为 36,37,38,...,53
%其中原点(县城)O 标号为 50
A=ones(n,n);
A=A*100000;
A=A-diag(diag(A));
%P 存储各点的前点
P=zeros(n,1);
%(A,36) (B,37) (C,38) (D,39) (E,40) (F,41) (G,42) (H,43) (I,44)
%(J,45) (K,46) (L,47) (M,48) (N,49) (O,50) (P,51) (Q,52) R(53)
%从图中给出相应点的直接距离
A(1,36)=10.3;A(1,37)=5.9; A(1,38)=11.2; A(1,50)=6;
A(2,3)=4.8; A(2,5)=8.3; A(2,50)=9.2;
A(3,38)=7.9;A(3,39)=8.2;
A(4,8)=20.4;A(4,39)=12.7;
A(5,6)=9.7; A(5,39)=11.3; A(5,48)=11.4;
A(6,7)=7.3; A(6,47)=11.8;A(6,48)=9.5;
A(7,39)=15.1; A(7,40)=7.2; A(7,47)=14.5;
A(8,40)=8;

A(9,40)=7.8; A(9,41)=5.6;
A(10,41)=10.8;
A(11,40)=14.2; A(11,42)=6.8; A(11,45)=13.2;
A(12,41)=12.2; A(12,43)=10.2;
A(13,14)=8.6; A(13,42)=8.6;A(13,44)=16.4;A(13,45)=9.8;
A(14,15)=15;A(14,43)=9.9;
A(15,44)=8.8;
A(16,17)=6.8; A(16,44)=11.8;
A(17,22)=6.7; A(17,46)=9.8;
A(18,44)=8.2;A(18,45)=8.2; A(18,46)=9.2;
A(19,20)=9.3; A(19,45)=8.1; A(19,47)=7.2;
A(20,21)=7.9;A(20,25)=6.5; A(20,47)=5.5;
A(21,23)=9.1; A(21,25)=7.8; A(21,46)=4.1;
A(22,23)=10; A(22,46)=10.1;
A(23,24)=8.9; A(23,49)=7.9;
A(24,27)=18.8;A(24,49)=13.2;
A(25,48)=12; A(25,49)=8.8;
A(26,27)=7.8; A(26,49)=10.5; A(26,51)=10.5;
A(27,28)=7.9;
```

```

A(28,51)=12.1; A(28,52)=8.3;
A(29,51)=15.2;A(29,52)=7.2; A(29,53)=7.9;
A(30,32)=10.3; A(30,52)=7.7;
A(31,32)=8.1; A(31,33)=7.3; A(31,53)=9.2;
A(32,33)=19; A(32,35)=14.9;
A(33,35)=20.3;A(33,36)=7.4;
A(34,35)=8.2; A(34,36)=11.5;
A(36,37)=12.2; A(36,38)=21.5; A(36,53)=8.8; A(36,50)=16.3;
A(37,38)=11;A(37,50)=11.9;
A(38,39)=16.1; A(38,50)=11.5;
A(48,49)=14.2;A(48,50)=19.8;
A(50,51)=10.1; A(50,53)=12.9;

for j=1:n
    for i=1:j-1
        A(j,i)=A(i,j); %使对称
    end
end
U=A; %原始距离矩阵

[m,n]=size(A);

B=zeros(m,n);
B=A;

%利用 Floyd 算法计算最短距离矩阵
for k=1:n
    for i=1 :n
        for j=1:n
            t=B(i,k)+B(k,j);
            if t<B(i,j) B(i,j)=t; end
        end
    end
end

%P 存储各点的前点
P=zeros(n,1);

for i=1:n
    mins=B(i,1)+B(1,50);
    k=50;
    for j=2:n
        s=B(i,j)+B(j,50);
        if (s<mins&&i~=j&&U(i,j)<1000) mins=s; k=j; end
    end %求得各点的前点
    P(i)=k;
end
fprintf('输出各点及前点:\n');
for i=1:n
    fprintf('i=%2d P=%2d\n',i,P(i));
end

```







```
*****
%问题 1,分成 3 组
%输出路线给定后的结果
%(A,36) (B,37) (C,38) (D,39) (E,40) (F,41) (G,42) (H,43) (I,44)
%(J,45) (K,46) (L,47) (M,48) (N,49) (O,50) (P,51) (Q,52) R(53)
Path1=[50,51,28,27,26,49,24,23,22,17,16,44,15,44,18,46,21,20,25,48,50];
%线路 2 原路线
%Path2=[50,2,5,6,47,19,45,11,42,13,14,43,12,41,10,41,9,40,7,40,8,4,39,3,38,50];
%线路 2 调整后路线
Path2=[50,2,5,6,40,8,40,9,41,10,41,12,43,14,13,42,11,45,19,47,6,5,2,50];
;
%线路 3 原路线
%Path3=[50,53,29,52,30,32,31,33,35,34,36,37,1,50];
%线路 3 调整后路线
Path3=[50,53,29,52,30,32,31,33,35,34,36,1,37,38,3,39,4,39,3,2,50];
L1=length(Path1);
L2=length(Path2);
L3=length(Path3);
fprintf('问题 1 结果:\n');

d1=0;
for i=1:L1-1
    d1=d1+B(Path1(i),Path1(i+1));
end
fprintf('路线 1 总距离:%6d\n',d1);

d2=0;
for i=1:L2-1
    d2=d2+B(Path2(i),Path2(i+1));
end
fprintf('路线 2 总距离:%6d\n',d2);

d3=0;
for i=1:L3-1
    d3=d3+B(Path3(i),Path3(i+1));
end
fprintf('路线 3 总距离:%6d\n',d3);

*****
%问题 2,分成 4 组
%输出路线给定后的结果
%(A,36) (B,37) (C,38) (D,39) (E,40) (F,41) (G,42) (H,43) (I,44)
%(J,45) (K,46) (L,47) (M,48) (N,49) (O,50) (P,51) (Q,52) R(53)
Bath1=[50,2,5,6,7,40,8,40,11,42,12,43,12,41,10,41,9,40,7,6,5,2,50];
Bath2=[50,53,29,52,30,52,28,27,26,49,24,23,22,17,16,17,46,22,23,49,26,51,50];
Bath3=[50,48,25,20,21,46,18,44,15,14,13,45,19,47,6,48,50];
Bath4=[50,53,36,33,31,32,35,34,37,1,38,3,39,4,39,3,2,50];
L1=length(Bath1);
```

```

L2=length(Bath2);
L3=length(Bath3);
L4=length(Bath4);

fprintf('\n 问题 2 结果:\n');
s1=0;
for i=1:L1-1
    s1=s1+B(Bath1(i),Bath1(i+1));
end
fprintf('路线 1 总距离:%6d\n',s1);

s2=0;
for i=1:L2-1
    s2=s2+B(Bath2(i),Bath2(i+1));
end
fprintf('路线 2 总距离:%6d\n',s2);

s3=0;
for i=1:L3-1
    s3=s3+B(Bath3(i),Bath3(i+1));
end
fprintf('路线 3 总距离:%6d\n',s3);

s4=0;
for i=1:L3-1
    s4=s4+B(Bath4(i),Bath4(i+1));
end
fprintf('路线 4 总距离:%6d\n',s4);

%各线路城镇数
N1=[4,4,5,5];

%各线路乡村数
N2=[9,10,8,8];
S=[s1,s2,s3,s4];
T=S/35+N1*2+N2;
fprintf('\n4 条线路所花时间%5.2f,%5.2f,%5.2f,%5.2f\n',T(1),T(2),T(3),T(4));

*****

%问题 3 解答
%求出到 O 点最远的点
P=B(50,:);
[u,k]=max(P);

if(k>35) t=2;end %镇停留时间
if(k<=35) t=1; end %村停留时间

t=t+u/35*2;
fprintf('O 到点%2d 距离为%5.2f 公里,所花时间为%5.2f 小时\n',k,u,t);

```





## 程序 2: MatLab 程序 B98n.m

该程序实现的功能为: 输出三条路线要经过的点构成的距离矩阵, 便于 Lingo 求解最优路线及最短距离。

```
%这里为篇幅所限, 省略与前面 B98.m 程序相同部分, 计算 53 个点的最短距离矩阵 B
%但网址: http://www.mathor.com/thread-11976-1-1.html 中对应程序是完全的。

%(A,36) (B,37) (C,38) (D,39) (E,40) (F,41) (G,42) (H,43) (I,44)
%(J,45) (K,46) (L,47) (M,48) (N,49) (O,50) (P,51) (Q,52) R(53)
%输出距离
Path1=[50,48,20,25,21,46,18,44,15,16,17,51,28,26,27,49,22,23,24];
L1=length(Path1);
D1=zeros(L1,L1);
for i=1:L1
    for j=1:L1
        D1(i,j)=B(Path1(i),Path1(j));
    end
end
%输出路线 1 对应的距离矩阵
fid1=fopen('d:\lingo12\dat\B98D1.txt','w');

for i=1:L1
    for j=1:L1
        fprintf(fid1,'%4.1f ',D1(i,j));
    end
    fprintf(fid1,'\r\n');
end
fclose(fid1);
%(A,36) (B,37) (C,38) (D,39) (E,40) (F,41) (G,42) (H,43) (I,44)
%(J,45) (K,46) (L,47) (M,48) (N,49) (O,50) (P,51) (Q,52) R(53)
Path2=[50,2,5,6,7,40,8,9,41,10,12,43,14,13,42,11,45,19,47];
L2=length(Path2);
D2=zeros(L2,L2);
for i=1:L2
    for j=1:L2
        D2(i,j)=B(Path2(i),Path2(j));
    end
end
%输出路线 2 对应的距离矩阵
fid2=fopen('d:\lingo12\dat\B98D2.txt','w');
for i=1:L2
    for j=1:L2
        fprintf(fid2,'%4.1f ',D2(i,j));
    end
    fprintf(fid2,'\r\n');
end
fclose(fid2);
```

```

%(A,36) (B,37) (C,38) (D,39) (E,40) (F,41) (G,42) (H,43) (I,44)
%(J,45) (K,46) (L,47) (M,48) (N,49) (O,50) (P,51) (Q,52) R(53)
Path3=[50,53,29,52,30,31,32,1,36,33,34,35,37,38,2,3,4];
L3=length(Path3);
D3=zeros(L3,L3);
for i=1:L3
    for j=1:L3
        D3(i,j)=B(Path3(i),Path3(j));
    end
end
%输出路线 3 对应的距离矩阵
fid3=fopen('d:\lingo12\dat\B98D3.txt','w');
for i=1:L3
    for j=1:L3
        fprintf(fid3,'%4.1f ',D3(i,j));
    end
    fprintf(fid3,'\r\n');
end
fclose(fid3);

```

### 程序 3: Lingo 程序 B98.lg4

该程序实现的功能为：给定要经过的点以及对应的距离矩阵后，走遍所有点的最优方式及最短路径。

```

!TSP question;
MODEL:
SETS:
point/1..19/:u;
link(point,point):d,x;
ENDSETS
DATA:
d=@file('d:\lingo12\dat\B98D1.txt'); !文件路径;
!只输出为 x(i,j)=1 的值;
@text()=@writefor(link(i,j)|x(i,j)#GT#0:'x(',i,',',j,')=',x(i,j),' ');
ENDDATA
MIN=@SUM(link:d*x);
n=@size(point);
@for(point(j):@sum(point(i)|j#ne#i:x(i,j))=1); !点 j 前有一个点相连;
@for(point(i):@sum(point(j)|j#ne#i:x(i,j))=1); !点 i 后有一个点相连;
@for(link(i,j)|i#NE#j#and#i#gt#1:u(i)-u(j)+n*x(i,j)<=n-1);
@FOR(link:@BIN(x));
end

```

该程序需要 B98n.m 生成的三个数据文件 B98D1.txt, B98D2.txt, B98D3.txt。B98D1.txt 点数为 19, 输出最优距离为 191.1km; B98D2.txt 点数为 19, 输出最优距离为 216.4km; B98D3.txt 点数为 17, 输出最优距离为 192.3km, 正是问题 1 调整后的计算结果。





### 4.1.3 状态转移与图论模型的巧妙结合

这里我们通过几个经典的智力问题，讲解状态转移与图论模型的巧妙结合。对这些问题，通常并不需要数学知识进行求解，但我们可以利用数学知识建立数学模型，然后转化为标准的图论模型进行求解。

#### 1. 人、狼、羊、菜渡河问题

一个摆渡人希望用一条小船把一只狼，一只羊和一篮白菜从一条河的左岸渡到右岸去，而船小只能容纳人，狼，羊，菜中的两个，决不能在无人看守的情况下留下狼和羊在一起，也不允许羊和白菜在一起，应怎样渡河才能将狼、羊、白菜都运过去？

采用试探法可以得到以下两种渡河方式，如表 4-6 所列。

表 4-6

方 法	渡河方式
方式 1	人、羊（去） → 人（回） → 人、狼（去） → 人、羊（回） → 人、菜（去） → 人（回） → 人、羊（去）
方式 2	人、羊（去） → 人（回） → 人、菜（去） → 人、羊（回） → 人、狼（去） → 人（回） → 人、羊（去）

表 4-6 中列出的渡河方式是通过试探法得到的，然而对于这样的问题，如何采用数学的方法来获得最优解呢？这是需要解决的问题，下面我们采用数学方法得到可行的渡河方式。

我们用  $(x_1, x_2, x_3, x_4)$  作为状态变量表示人、狼、羊、菜在此岸的状态，若  $x_1 = 0$  表示人在彼岸， $x_1 = 1$  表示人在此岸。如  $(1, 0, 1, 0)$  表示人和羊在此岸，狼和菜在彼岸。根据问题的要求，我们知道总共有 10 个状态是安全的。它们集合  $S$  为：

$$S = \{(1, 1, 1, 1), (1, 1, 1, 0), (1, 1, 0, 1), (1, 0, 1, 1), (1, 0, 1, 0), \\ (0, 0, 0, 0), (0, 0, 0, 1), (0, 0, 1, 0), (0, 1, 0, 0), (0, 1, 0, 1)\}$$

所有状态之间转移关系可以通过图 4-4 的状态转移图表示。如状态  $(1, 1, 1, 1)$  转化为  $(0, 1, 0, 1)$ ，表示人将羊运到对岸，反之也可以。

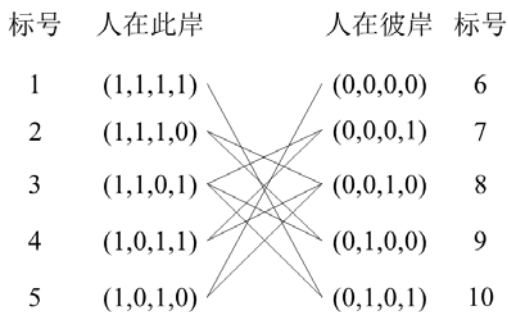


图 4-4

重新按照起始状态到目标状态的连接图如图 4-5 所示。由图 4-5 容易看出, 从初始状态  $(1,1,1,1)$  到目标状态  $(0,0,0,0)$  的最短路线共有两条, 都为 7 步, 而且对应的解恰好为前面试探法得到的解。

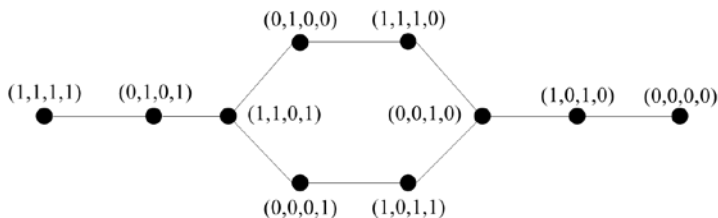


图 4-5

## 2. 商人过河问题

有 3 名商人各带一个仆人乘船渡河, 小船只能容纳两个人, 由他们自己划船。仆人们约定, 在河的人一岸, 一旦仆人的数量比商人多, 就杀人越货。但是如何乘船的大权掌握在商人们手里。问商人们怎样才能安全渡河?

### ◆ 问题分析

安全渡河问题是一个多步决策问题。每一步, 即船由此岸驶向彼岸或从彼岸回到此岸, 都要对船上的商人和仆人的个数作出决策。在保证安全的前提下, 在有限步内使全部人员过河。采用状态变量表示某一岸的人员状况, 决策变量表示船上人员状况。可以找出状态随决策变化的规律, 问题转化为在状态允许范围内(安全渡河的条件), 确定每一步的决策, 达到安全渡河的条件。

### ◆ 模型建立

设第  $k$  次渡河前此岸的商人数为  $x_k$ , 仆人数为  $y_k$ ,  $k=1,2,\dots$ 。因此状态变量为  $(x_k, y_k)$ , 其中  $x_k, y_k$  取值为 0,1,2,3。安全渡河条件下的状态集合为允许的状态集合, 记作  $S$ 。容易知道该集合为:

$$S=\{(0,0),(0,1),(0,2),(0,3),(3,0),(3,1),(3,2),(1,1),(2,2),(3,3)\} \quad (1)$$

$S$  共 10 种状态。每种安全状态既要满足此岸安全, 同时彼岸也要安全。

记第  $k$  次渡河船上的商人数为  $u_k$ , 仆人数为  $v_k$ 。则决策变量为  $d_k=(u_k, v_k)$ 。允许的决策集合记为  $D$ 。由船的容量可知  $D$  的集合为:

$$D=\{(2,0), (0,2), (1,1), (1,0), (0,1)\} \quad (2)$$

每次有 5 种决策可供选择。

由于  $k$  为奇数时船从此岸到彼岸,  $k$  为偶数时船从彼岸到此岸, 因此状态  $s_k$  随决策  $d_k$  而变化的规律为:

$$\begin{cases} s_0=(3,3) \\ s_k=s_{k-1}+(-1)^k \cdot d_k \quad k=1,2,3,\dots \end{cases} \quad (3)$$



(3) 式称为状态转移律。这样制定安全的渡河方案归结为如下的多步决策模型:

求决策  $d_k (k=1,2,3\cdots)$ , 使状态  $s_k \in S$  按照转移律 (3), 由初始状态  $(3,3)$ , 经过有限步  $n$  到达目的状态  $(0,0)$ , 并且步数  $n$  尽量小。

◆ 模型求解

(1) 手工求解。

对该问题, 我们可以采用图解的方法简单地用手工方法求解。

在如图 4-6 所示的  $xOy$  平面上, 作 3 行 3 列的格子。每个格子点  $(x,y)$  代表一种此岸状态。允许的 10 个状态在图上用黑点标出。每次决策或者是过去或者是回来。过去时采用向下走 1 个  $[d=(0,1)]$  或 2 个格子  $[d=(0,2)]$ , 或者向左走 1 个  $[d=(1,0)]$  或 2 个格子  $[d=(2,0)]$ , 或者向左下走两个格子  $[d=(1,1)]$ 。回来时采用向上走 1 个  $[d=(0,1)]$  或 2 个格子  $[d=(0,2)]$ , 或者向右走 1 个  $[d=(1,0)]$  或 2 个格子  $[d=(2,0)]$ , 或者向右上走两个格子  $[d=(1,1)]$ 。每过去一次同时需要回来一次, 直到从初始状态  $(3,3)$  点到达  $(0,0)$  状态为止。

求解过程如同下棋, 过程见图 4-6 标出的箭头方向。

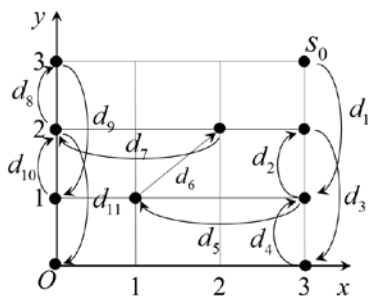


图 4-6

(2) 图论求解。

前面的方法采用的是如下棋一样的手工操作。通常只能找到可行解, 不能保证是最优解, 而且通常不具有一般性。这里我们采用图论的方法进行一般求解。

由图 4-6 知, 此岸共有 10 种状态。每种状态用二维向量  $(x,y)$  来表示。我们考虑每种状态可以转化的其他状态, 由此构成一个图。但一种状态能否转化为另一种状态, 不但跟该状态是否安全有关, 而且跟船在此岸还是在彼岸有关。如  $(3,1)$  状态, 当船在此岸时 (偶数次渡河后), 可以转化为  $(3,0)$ ; 当船在彼岸时 (奇数次渡河后) 就不能转化为  $(3,0)$ 。因此, 为统一考虑状态之间的转化, 我们采用三维向量  $(x,y,z)$  来表示。  $x$  表示此岸商人数 ( $x=0,1,2,3$ ),  $y$  表示此岸仆人数  $y=0,1,2,3$ ,  $z$  表示船在此岸还是彼岸 ( $z=1$  表示船在此岸,  $z=0$  表示船在彼岸)。因此总共有 20 个状态, 我们可以建立 20 个顶点的图。

根据船每次最多装两个人, 容易建立状态之间的转移关系。我们容易得到 20 种状态之间的转移关系如图 4-7 所示。该图对 20 个状态分别标号, 同时给出了各状态之间的转移。两个状态之间的转移是相互的。该图为无向图。对该图进行重新用标号进行连接, 得到如图 4-8 所示的连接图 (相连的两点表示两状态可互相转化)。我们的问题转化为求顶点 1 到





顶点 20 的最短路。显然由图 4-8 可知, 最短路为  $1 \rightarrow 13 \rightarrow 2 \rightarrow 14 \rightarrow 3 \rightarrow 16 \rightarrow 5 \rightarrow 18 \rightarrow 7 \rightarrow 19 \rightarrow 8 \rightarrow 20$  或  $1 \rightarrow 15 \rightarrow 2 \rightarrow 14 \rightarrow 3 \rightarrow 16 \rightarrow 5 \rightarrow 18 \rightarrow 7 \rightarrow 19 \rightarrow 8 \rightarrow 20$ 。其最短路长度为 11。即最少经过 11 步可以由(3,3)转移到(0,0), 完成安全渡河。

用状态向量表示的如下 (只表示此岸状态):  $(3,3) \rightarrow (3,1) \rightarrow (3,2) \rightarrow (3,0) \rightarrow (3,1) \rightarrow (1,1) \rightarrow (2,2) \rightarrow (0,2) \rightarrow (0,3) \rightarrow (0,1) \rightarrow (0,2) \rightarrow (0,0)$ , 该顺序与跟 1 中得到的  $s_0, s_1, s_2, \dots, s_{11}$  完全一样。或者为  $(3,3) \rightarrow (2,2) \rightarrow (3,2) \rightarrow (3,0) \rightarrow (3,1) \rightarrow (1,1) \rightarrow (2,2) \rightarrow (0,2) \rightarrow (0,3) \rightarrow (0,1) \rightarrow (0,2) \rightarrow (0,0)$ 。共有两种步数最少的渡河方法。

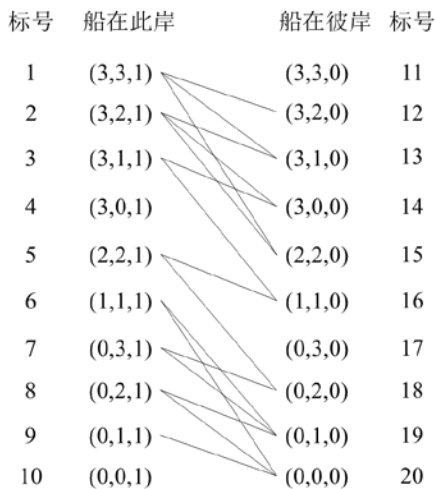


图 4-7

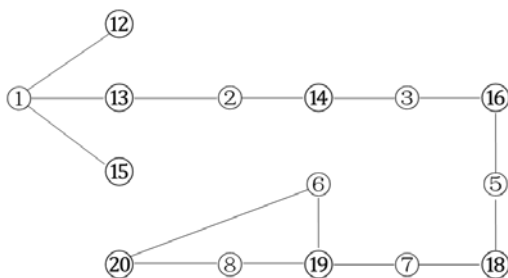


图 4-8

### 3. 等分酒问题

现有一只装满 8 斤酒的瓶子和两只分别装 5 斤和 3 斤酒的空瓶, 如何才能将这 8 斤酒分成两等份?

手工操作法 设状态向量(a,b,c), 其中 a 代表可装 8 斤酒的瓶子, b 代表可装 5 斤酒的瓶子, c 代表可装 3 斤酒的瓶子。该问题转化为如何将初始状态(8,0,0)经过转化达到目标状态(4,4,0)。



并且其操作过程必须满足条件：任何两瓶之间操作必须满足其中一个瓶子清空为 0 或另一个装满。

我们可以采用状态转移的方法，利用图论知识进行求解。从初始状态(8,0,0)开始，依此推出新出现的状态，由新出现的状态再推出更新的状态，直到不能推出新状态为止。分酒问题的状态转移图如图 4-9 所示。

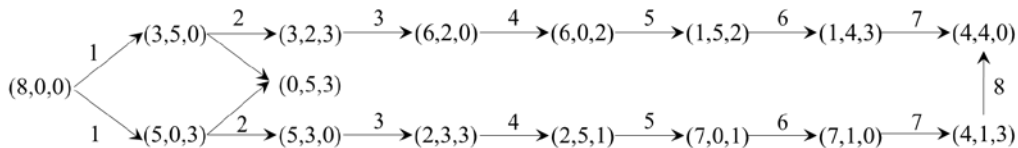


图 4-9

从图 4-9 中容易看出，从(8,0,0)状态可以有两种方式实现平分酒。一种方式是采用上面的方式，通过 7 步实现。另一种方式是下面的方式，通过 8 步实现。当然采用上面的方式是最少步数的方式。实现过程为

$$(8,0,0) \rightarrow (3,5,0) \rightarrow (3,2,3) \rightarrow (6,2,0) \rightarrow (6,0,2) \rightarrow (1,5,2) \rightarrow (1,4,3) \rightarrow (4,4,0)$$

#### 4.1.4 最优树问题及Lingo求解

树 连通且不含圈的无向图称为树，常用  $T$  表示。树中的边称为树枝，树中为 1 的顶点称为树叶。树的示例如图 4-10 所示。

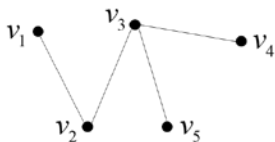


图 4-10

生成树 若  $T$  是包含图  $G$  的全部顶点的子图，它又是树，则称  $T$  是  $G$  的生成树。

最小生成树 设  $T = (V, E_1)$  是赋权图  $G = (V, E)$  的一棵生成树，称  $T$  中全部边上的权数之和为生成树的权，记为  $w(T)$ ，即  $w(T) = \sum_{e \in E_1} w(e)$ 。

如果生成树  $T^*$  的权  $w(T^*)$  是  $G$  的所有生成树的权中最小者，则称  $T^*$  是  $G$  的最优树，即  $w(T^*) = \min_T \{w(T)\}$ ，式中取遍  $G$  的所有生成树  $T$ 。

##### 1. 模型简介

在许多实际问题中，如在许多城市间建立公路网、输电网或通信网络，都可以归结为赋权图的最优树问题。如在一个城市中，对若干个居民点要供应自来水，已经预算出连接各点间的直接管道的造价，要求给出一个总造价最小的铺设方案等。



图论中最优树的求解通常有两种算法：Kruskal 算法（或避圈法）和 Prim 算法（破圈法）。这里我们给出利用 Lingo 求解最优树的方法。

设无向图共有  $n$  个节点，其赋权图的邻接矩阵为  $\mathbf{d}_{n \times n}$ ， $d_{ij}$  表示节点  $i$  到  $j$  的距离。 $\mathbf{d}$  为对称矩阵，且令  $d_{ii} = 0$ 。现求根节点 1 到各节点生成的最优树，要求各线路上的权值和最小。其线性规划模型为决策变量：

$$\text{设 } x_{ij} = \begin{cases} 1 & \text{节点 } i \text{ 与节点 } j \text{ 连通} \\ 0 & \text{节点 } i \text{ 与节点 } j \text{ 不连通} \end{cases}$$

目标函数为寻找一条从起始点 1 到各节点生成的最优树，要求各线路上的权值和最小，故目标函数为：

$$\min Z = \sum_{i=1}^n \sum_{j=1}^n d_{ij} \cdot x_{ij}$$

(1) 对起始点 1 至少有一条路出去，故有：

$$\sum_{j=2}^n x_{1j} \geq 1$$

(2) 对其余各节点，恰有一条路进入，故有：

$$\sum_{\substack{k=1 \\ k \neq i}}^n x_{ki} = 1$$

(3) 所有节点不出现圈，约束为：

$$u_i - u_j + n \cdot x_{ij} \leq n - 1 \quad i, j = 1, 2, \dots, n$$

总的线性规划模型为：

$$\begin{aligned} \min Z &= \sum_{i=1}^n \sum_{j=1}^n d_{ij} \cdot x_{ij} \\ \text{s.t.} &\begin{cases} \sum_{j=2}^n x_{1j} \geq 1 \\ \sum_{\substack{k=1 \\ k \neq i}}^n x_{ki} = 1 & i = 2, 3, \dots, n \\ u_i - u_j + n x_{ij} \leq n - 1 & i, j = 1, 2, \dots, n \\ x_{ij} = 0 \text{ 或 } 1 \end{cases} \end{aligned}$$

## 2. 案例分析——管道铺设问题

某地 10 个城镇地理位置示意图见图 4-11 所示，它们之间的距离见表 4-7 所示。城镇





①处有一条河流，现需要从各城镇之间铺设管道，使城镇①处的水可以输送到各城镇，求铺设管道最少的设计方式。

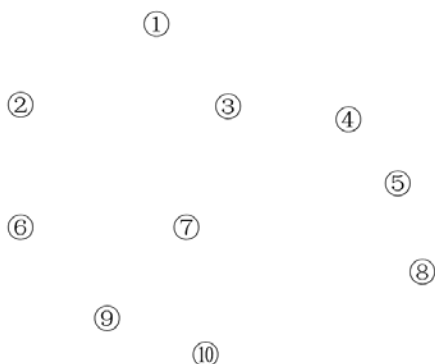


图 4-11

表 4-7

城镇 距离 城镇	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩
①	0	8	5	9	12	14	12	16	17	22
②	8	0	9	15	16	8	11	18	14	22
③	5	9	0	7	9	11	7	12	12	17
④	9	15	7	0	3	17	10	7	15	15
⑤	12	16	9	3	0	8	10	6	15	15
⑥	14	8	11	17	8	0	9	14	8	16
⑦	12	11	7	10	10	9	0	8	6	11
⑧	16	18	12	7	6	14	8	0	11	11
⑨	17	14	12	25	15	8	6	11	0	10
⑩	22	22	17	15	15	16	11	11	10	0

该问题实际上是求从点 1 出发的最优树问题。其 Lingo 实现程序为：

```
! 最优树的 Lingo 程序;
model:
sets:
point/1..10/:u;
link(point,point):d,x;

endsets
data:
d=0,8,5,9,12,14,12,16,17,22,
  8,0,9,15,16,8,11,18,14,22,
  5,9,0,7,9,11,7,12,12,17,
  9,15,7,0,3,17,10,7,15,15,
  12,16,9,3,0,8,10,6,15,15,
  14,8,11,17,8,0,9,14,8,16,
```



```

12,11,7,10,10,9,0,8,6,11,
16,18,12,7,6,14,8,0,11,11,
17,14,12,25,15,8,6,11,0,10,
22,22,17,15,15,16,11,11,10,0;
@text()=@writefor(link(i,j)|x(i,j)#GT#0:'x(',i,',','j,')=' ,x(i,j),' ');
enddata
min=@sum(link(i,j)|i#ne#j:d(i,j)*x(i,j));
n=@size(point);
@sum(point(j)|j#gt#1:x(1,j))>=1; !从起始点出来至少1条路;
@for(point(i)|i#ne#1:@sum(point(j)|j#ne#i:x(j,i))=1); !除起始点外,每点只有一
路进入;
@for(link(i,j):@bin(x(i,j)));
@for(link(i,j)|i#ne#j:u(i)-u(j)+n*x(i,j)<=n-1); !不构成圈;
end

```

结果为  $\min Z=60$ ,  $x(1,2)=1$   $x(1,3)=1$   $x(3,4)=1$   $x(4,5)=1$   $x(9,6)=1$   $x(3,7)=1$   
 $x(7,9)=1$   $x(5,8)=1$   $x(9,10)=1$ , 故最优树（最佳铺设管道的方式）见图 4-12 所示。

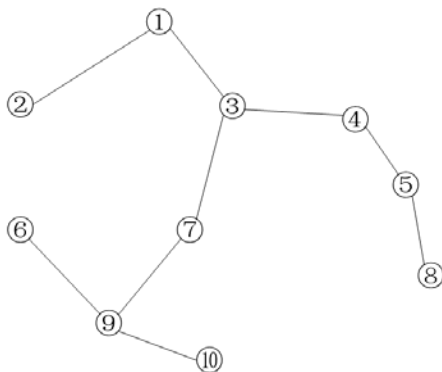


图 4-12

### 4.1.5 竞赛图与循环比赛排名问题

在图论中，完全图的定向图称为竞赛图。在实际问题中，可用于球队比赛排名，论文被引用次数的排名等。若干支球队参加单循环比赛，各队两两交锋。假设每场比赛只计胜负，不计比分，在比赛结束后排名次。

设  $n$  支球队或队员比赛，第  $i$  支球队与第  $j$  支球队由比赛表现的能力表示为

$$a_{ij} = p_{ij}, a_{ji} = 1 - p_{ij}, (i = 1, 2, \dots, n-1; j = i+1)$$

其中  $p_{ij}$  表示第  $i$  支球队胜第  $j$  支球队的概率，且设  $a_{ii} = 0$ ，则第  $i$  支球队胜其余  $n-1$  支球队的能力表示为：

$$s_i = \sum_{j=1}^n a_{ij} \quad (i = 1, 2, \dots, n)$$



则各球队的排名根据  $\{s_i\}$  的大小进行。 $s_i$  越大越靠前, 越小越靠后。

但实际中  $p_{ij}$  并不知道, 只有根据比赛进行估计。

(1) 当进行  $m$  场比赛, 第  $i$  支球队胜  $l$  次, 则估计  $p_{ij} = \frac{l}{m}$ ,  $p_{ji} = \frac{m-l}{m}$ 。

(2) 当只进行一次比赛时, 若第  $i$  支球队胜, 则记  $a_{ij} = 1$ ,  $a_{ji} = 0$ ; 若第  $j$  支球队胜, 则记  $a_{ij} = 0$ ,  $a_{ji} = 1$ 。

本节我们仅对只进行一次比赛的情况展开讨论。

### 1. 双向连通竞赛图

对于任何一对顶点, 存在两条有向路径, 使两顶点可以互相连通, 这种有向图称为双向连通竞赛图, 其示意图如图 4-13 所示。

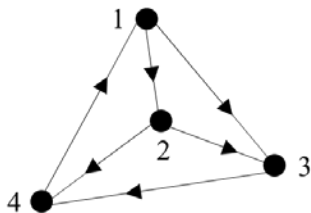


图 4-13

图 4-13 是 4 个队比赛结果的双向连通竞赛图。其对应的邻接矩阵为:

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

设  $e = (1, 1, 1, 1)^T$ , 则  $s_1 = Ae = (2, 2, 1, 1)$ 。该式表明各队胜的场次数。

$s_2 = As_1 = (3, 2, 1, 1)$ 。该式是各人的 2 级得分, 其意义是他战胜的各个球队的得分之和。

可以作为排名的依据, 但无法排出 3 和 4 的名次, 可继续进行下去, 得到结果如下:

$$s_3 = As_2 = (3, 3, 2, 3), \quad s_4 = As_3 = (5, 5, 3, 3), \quad s_5 = As_4 = (8, 6, 3, 5)$$

$$s_6 = As_5 = (9, 8, 5, 8) \quad s_7 = As_6 = (13, 13, 8, 9) \quad s_8 = As_7 = (21, 17, 9, 13)$$

$s_k$  各分量代表各人的第  $k$  级得分, 其意义是他战胜的各个球队的前一级得分之和, 因此更可以作为排名的依据。我们容易得出其排名为: 1->2->4->3。

对一般性, 记  $s_1 = Ae$ ,  $s_2 = As_1, \dots, s_k = As_{k-1}$ 。则有

$$s_k = As_{k-1} = A^k \cdot e \quad (k = 1, 2, \dots)$$

当迭代次数越多,名次排定顺序越稳定。可将其较高级的得分作为排名的依据。对其他双向连通竞赛图也可以采用类似方法迭代计算得到。

但这里有一个问题,是否双向连通竞赛图可以排出确定的名次,另外是否还有更简单的方法?为了回答这个问题,我们先给出素阵的定义。

素阵 对于  $n(n \geq 4)$  个顶点的双向连通竞赛图的邻接矩阵  $A$ , 一定存在正整数  $r$ , 使得  $A^r > 0$ , 这样的  $A$  就称为素阵。

Perron—Frobenius 定理:

素阵  $A$  的最大特征根为正单根  $\lambda$ ,  $\lambda$  对应正特征向量  $s$ , 且有

$$\lim_{k \rightarrow \infty} \frac{A^k \cdot e}{\lambda^k} = s$$

说明  $k$  级得分分量  $s_k$ , 当  $k \rightarrow \infty$  时(归一化后)将趋向于  $A$  的对应最大特征根的特征向量  $s$ 。因此特征向量  $s$  可作为排名次依据的得分向量。

我们求出前面求矩阵  $A$  的最大特征值及对应特征向量有:

$\lambda_{\max} = 1.3953$ , 对应特征向量为  $(0.6256, 0.5516, 0.3213, 0.4484)$ , 特征向量中分量大小与排名次序一致。

## 2. 非双向连通竞赛图

对于非双向连通竞赛图, 则没有此结论, 如:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

对应的非双向连通竞赛图示意图见图 4-14 所示。

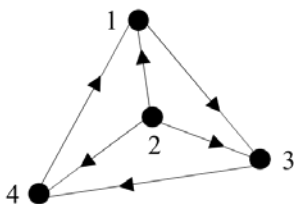


图 4-14

按照前面的方法计算得到:

$$s_1 = A \cdot e = (1, 3, 1, 1), \quad s_2 = A \cdot s_1 = (1, 3, 1, 1), \quad s_3 = A \cdot s_2 = (1, 3, 1, 1)$$

其最大特征值对应特征向量为  $(0.2887, 0.8660, 0.2887, 0.2887)$ 。

从结果看无法对 1, 3, 4 进行排名。





## 3. 足球比赛排名问题 (CUMCM1993B)

下表给出了我国 12 支足球队在 1988—1989 年足球甲级队联赛中的成绩, 要求:

- (1) 设计一个依据这些成绩排出诸队名次的算法, 并给出用该算法排名次的结果;
- (2) 把算法推广到任意  $N$  个队的情况;
- (3) 讨论数据应具备什么样的条件, 用你的方法才能够排出诸队的名次。

常用比赛结果表如表 4-8 所列。

表 4-8

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
T1	×	0:1 1:0 0:0	2:2 1:0 0:2	2:0 3:1 1:0	3:1	1:0	0:1 1:3	0:2 2:1	1:0 4:0	1:1 1:1	×	×
T2		×	2:0 0:1 1:3	0:0 2:0 0:0	1:1	2:1	1:1 1:1	0:0 0:0	2:0 1:1	0:2 0:0	×	×
T3			×	4:2 1:1 0:0	2:1	3:0	1:0 1:4	0:1 3:1	1:0 2:3	0:1 2:0	×	×
T4				×	2:3	0:1	0:5 2:3	2:1 1:3	0:1 0:0	0:1 1:1	×	×
T5					×	0:1	×	×	×	×	1:0 1:2	0:0 1:1
T6						×	×	×	×	×	×	×
T7							×	1:0 2:0 0:0	2:1 3:0 1:0	3:1 3:0 2:2	3:1	2:0
T8								×	0:1 1:2 2:0	1:1 1:0 0:1	3:1	0:0
T9									×	3:0 1:0	1:0	1:0
T10										×	1:0	2:0
T11											×	1:1 1:2 1:1
T12												×

说明: (1) 12 支球队依次记作 T1, T2, ...T12。

(2) 符号 × 表示两队未曾比赛。

(3) 数字表示两队比赛结果, 如 T3 行与 T8 列交叉处的数字表示: T3 与 T8 比赛了 2 场; T3 与 T8 的进球数之比为 0:1 和 3:1。

◆ 建立邻接矩阵  $A$

对于  $i$  队和  $j$  队而言, 若  $i$  队胜  $j$  队的场次多, 则令  $a_{ij}=1, a_{ji}=0$ ; 若  $i$  队和  $j$  队胜的场次一样, 但  $i$  队比  $j$  队的净胜球多, 则令  $a_{ij}=1, a_{ji}=0$ ; 若  $i$  队和  $j$  队胜的场次一样, 净胜



球也一样, 或者  $i$  队和  $j$  队没有交战, 则令  $a_{ij} = -1, a_{ji} = -1$ , 表示  $i$  队和  $j$  队的胜败待定。由此得到尚未完善的邻接矩阵  $A$  如表 4-9 所列。

表 4-9

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
T1	0	-1	0	1	1	1	0	0	1	-1	-1	-1
T2	-1	0	0	1	0	1	0	-1	1	0	-1	-1
T3	1	1	0	1	1	1	0	1	-1	1	-1	-1
T4	0	0	0	0	0	0	0	0	0	0	-1	-1
T5	0	-1	0	1	0	0	-1	-1	-1	-1	-1	0
T6	0	0	0	1	1	0	-1	-1	-1	-1	-1	-1
T7	1	-1	1	1	-1	-1	0	1	1	1	1	1
T8	1	-1	0	1	-1	-1	0	0	-1	-1	1	-1
T9	0	0	-1	1	-1	-1	0	-1	0	1	1	1
T10	-1	1	0	1	-1	-1	0	-1	0	0	1	1
T11	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0
T12	-1	-1	-1	-1	1	-1	0	-1	0	0	1	0

◆ 修正邻接矩阵  $A$

计算各队一级和二级得分:

$$a_1 = (4, 3, 7, 0, 1, 2, 8, 3, 4, 4, 0, 2)$$

$$a_2 = (7, 6, 17, 0, 0, 1, 24, 4, 6, 5, 0, 1)$$

其中一级得分为各队所胜队数, 二级得分为各队所胜队的一级得分和。

根据一级得分修正邻接矩阵  $A$  中尚未确定的比赛胜败, 当  $a_{ij} = -1, a1_i > a1_j$  时, 令  $a_{ij} = 1, a_{ji} = 0$ , 表示  $i$  队胜  $j$  队。经过一级得分修正  $A$  后, 对尚未确定胜败的队, 采用二级得分修正, 当  $a_{ij} = -1, a2_i > a2_j$  时, 令  $a_{ij} = 1, a_{ji} = 0$ , 表示  $i$  队胜  $j$  队。

对尚未确定胜败的队伍, 采用抽签(随机)方式确定胜败。得到修正后的邻接矩阵见表 4-10 所列。

表 4-10

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12
T1	0	1	0	1	1	1	0	0	1	1	1	1
T2	0	0	0	1	1	1	0	1	1	0	1	1
T3	1	1	0	1	1	1	0	1	1	1	1	1
T4	0	0	0	0	0	0	0	0	0	0	0	0
T5	0	0	0	1	0	0	0	0	0	0	1	0
T6	0	0	0	1	1	0	0	0	0	0	1	0
T7	1	1	1	1	1	1	0	1	1	1	1	1
T8	1	0	0	1	1	1	0	0	0	0	1	1
T9	0	0	0	1	1	1	0	1	0	1	1	1
T10	0	1	0	1	1	1	0	1	0	0	1	1
T11	0	0	0	1	0	0	0	0	0	0	0	0
T12	0	0	0	1	1	1	0	0	0	0	1	0





## ◆ 排名

根据修正后的邻接矩阵  $A$ , 计算其各级得分或最大特征值对应的特征向量, 作为排名的依据。如 12 个队的 20 级得分为:

$$a_{20}=(3336894,2072949,6087933,0,0,0,9354474,1790388,2072949,2072949,0,0)$$

邻接矩阵  $A$  的最大特征值为 1.8637, 对应的特征向量为:

$$w=(0.1246,0.0774,0.2273,0,0,0,0.3492,0.0668,0.0774,0.0774,0,0)$$

最大特征值对应的特征向量与 12 个队的 20 级得分大小顺序一致, 可以作为排名的依据。

从特征向量来看, T7 排第 1, T3 排第 2, T1 排第 3; T2, T9, T10 排名相同, 居第 4 至第 6; T8 排第 7; T4, T5, T6, T11, T12 排名相同, 居第 8 至第 12; 对根据特征向量排名相同的队伍, 我们需要根据其他信息进行确定, 考察 T9, T10, T2。

由于一级得分  $a_1=(4,3,7,0,1,2,8,3,4,4,0,2)$ , 其中 T2, T9, T10 都交战 8 场, T2 胜 3 场, T9 和 T10 胜 4 场, 因此 T9 和 T10 排在 T2 前。再从二级得分来看, T9 得分为 6, T10 得分为 5。因此 T9 应排在 T10 前。因此 T2, T9, T10 的排名顺序为: T9, T10, T2。

考察 T4, T5, T6, T11, T12。由于一级得分 T6 和 T12 最高, 都为 2, 二级得分都为 1。因此 T6 和 T12 靠前。T5 的一级得分为 1, T11 的一级得分为 0, 则 T5 排在 T11 前。T11 和 T4 的一级和二级得分都为 0, 但从比赛成绩来看, T11 比赛 6 场, 平 1 场, 输 5 场, 而 T4 赛 9 场输 9 场, 因此 T11 排在 T4 前, T4 排最后。再考虑 T6 和 T12, 他们都各胜两场, 除都胜了 T5 外, T12 胜了 T11, 而 T6 胜了排名最后的 T4, 因此将 T12 排在 T6 前。因此 T4, T5, T6, T11, T12 的排名顺序为: T12, T6, T5, T11, T4。

综上所述, 最后得到的总排名为:

T7, T3, T1, T9, T10, T2, T12, T6, T5, T11, T4。

MatLab 实现程序:

```
%不完全的邻接矩阵 A
A=[0,-1,0,1,1,1,0,0,1,-1,-1,-1;
   -1,0,0,1,0,1,0,-1,1,0,-1,-1;
   1,1,0,1,1,1,0,1,-1,1,-1,-1;
   0,0,0,0,0,0,0,0,0,0,-1,-1;
   0,-1,0,1,0,0,-1,-1,-1,-1,-1,0;
   0,0,0,1,1,0,-1,-1,-1,-1,-1,-1;
   1,-1,1,1,-1,-1,0,1,1,1,1,1;
   1,-1,0,1,-1,-1,0,0,-1,-1,1,-1;
   0,0,-1,1,-1,-1,0,-1,0,1,1,1;
   -1,1,0,1,-1,-1,0,-1,0,0,1,1;
   -1,-1,-1,-1,-1,-1,0,0,0,0,0,0;
   -1,-1,-1,-1,1,-1,0,-1,0,0,1,0];
[m,n]=size(A);
D=A;
D(D(:) == -1)=2;
```

```

%获得一级得分向量
a1=zeros(1,n);
for i=1:m
    a1(i) = sum(sum(A(i,:)==1));
end

%获得二级得分向量
a2=zeros(1,n);
for i=1:m
    s=0;
    for j=1:n
        if(A(i,j)==1) s=s+a1(j); end
    end
    a2(i)=s;
end

%根据一级和二级得分向量完善邻接矩阵 A
for i=1:m
    for j=1:n
        if(A(i,j) == -1)
            if(a1(i)>a1(j)) A(i,j)=1; A(j,i)=0; end
            if(a1(i)<a1(j)) A(i,j)=0; A(j,i)=1; end
        end
    end
end

for i=1:m
    for j=1:n
        if(A(i,j) == -1)
            if(a2(i)>a2(j)) A(i,j)=1; A(j,i)=0; end
            if(a2(i)<a2(j)) A(i,j)=0; A(j,i)=1; end
        end
    end
end

for i=1:m
    for j=1:n
        if(A(i,j)==-1)
            r=rand(1,1);
            if(r>=0.5) A(i,j)=1; A(j,i)=0;
            else A(i,j)=0; A(j,i)=1; end
        end
    end
end

num=20;
Y=ones(n,1);
B=A;

```





```

for i=1:num
    Y=A*Y;
    B=B*A;
end
[u,v]=eig(A);
for i=1:n
    z(i)=v(i,i);
end
[p,k]=max(z) %获得最大特征值及位置

w=u(:,k); %获得最大特征值对应特征向量
w=w/sum(w);
fprintf('序号 得分 特征向量\n');
for k=1:n
    fprintf('%2d %-7d %-5.3f\n',k,Y(k),w(k));
end

```

输出结果为：

序号	得分	特征向量
1	3336894	0.125
2	2072949	0.077
3	6087933	0.227
4	0	0.000
5	0	0.000
6	0	0.000
7	9354474	0.349
8	1790388	0.067
9	2072949	0.077
10	2072949	0.077
11	0	0.000
12	0	0.000

## 4.2 排队论模型

排队论又称随机服务系统，它应用于一切服务系统，包括生产管理系统、通信系统、交通系统、计算机存储系统。它通过建立一些数学模型，以对随机发生的需求提供服务的系统预测。现实生活中如排队买票、病人排队就诊、轮船进港、高速路上汽车排队通过收费站、机器等待修理等都属于排队论问题。在历年数模竞赛中，排队论模型应用的全国数模竞赛中 2009B 的眼科病床的合理安排问题，美国数模竞赛中 2005B 收费站最佳配置问题。

本节内容分为三部分：排队论基本构成与指标，排队论的四种重要模型，排队论的计算机模拟。每部分内容都提供相应的计算程序，有的采用 Lingo 编制，有的采用 MatLab 编制。

## 4.2.1 排队论基本构成与指标

### 1. 排队论的基本构成

#### ◆ 输入过程

输入过程是描述顾客按照怎样的规律到达排队系统的。包括①顾客总体：顾客的来源是有限的还是无限的。②到达的类型：顾客是单个到达还是成批到达。③相继顾客到达的时间间隔：通常假定是相互独立分布，有的是等间隔到达，有的是服从负指数分布，有的是服从  $k$  阶 Erlang 分布。

#### ◆ 排队规则

排队规则指顾客按怎样的规定次序接受服务。常见的有等待制，损失制，混合制，闭合制。当一个顾客到达时所有服务台都不空闲，则此顾客排队等待直到得到服务后离开，称为等待制。在等待制中，可以采用先到先服务，如排队买票；也有后到先服务，如天气预报；也有随机服务，如电话服务；也有有优先权的服务，如危重病人可优先看病。当一个顾客到来时，所有服务台都不空闲，则该顾客立即离开不等待，称为损失制。顾客排队等候的人数是有限长的，称为混合制度。当顾客对象和服务对象相同且固定时是闭合制。如几名维修工人固定维修某个工厂的机器就属于闭合制。

#### ◆ 服务机构

服务机构主要包括：服务台的数量；服务时间服从的分布。常见的有定长分布、负指数分布、几何分布等。

### 2. 排队系统的数量指标

#### ◆ 队长与等待队长

队长（通常记为  $L_s$ ）是指系统中的平均顾客数（包括正在接受服务的顾客）。等待队长（通常记为  $L_q$ ）指系统中处于等待的顾客的数量。显然，队长等于等待队长加上正在服务的顾客数。

#### ◆ 等待时间

等待时间包括顾客的平均逗留时间（通常记为  $W_s$ ）和平均等待时间（通常记为  $W_q$ ）。顾客的平均逗留时间是指顾客进入系统到离开系统这段时间，包括等待时间和接受服务的时间。顾客的平均等待时间是指顾客进入系统到接受服务这段时间。

#### ◆ 忙期

从顾客到达空闲的系统，服务立即开始，直到再次变为空闲，这段时间是系统连续繁忙的时期，称之为系统的忙期。它反映了系统中服务机构工作强度，是衡量服务系统利用效率的指标，即

$$\text{服务强度} = \text{忙期} / \text{服务总时间} = 1 - \text{闲期} / \text{服务总时间}$$

闲期与忙期对应的系统的空闲时间，也就是系统连续保持空闲的时间长度。

计算这些指标的基础是表达系统状态的概率。所谓系统的状态是指系统中的顾客数，





如果系统中有  $n$  个顾客就说系统的状态是  $n$ ，它可能的数值是：

- ① 队长没有限制时， $n=0,1,2,\cdots$ ，
- ② 队长有限制，最大数为  $n$ ，则  $n=0,1,2,\cdots,N$
- ③ 即时制，服务台个数为  $C$  时， $n=0,1,2,\cdots,C$ 。该状态又表示正在工作的服务台数。

### 3. 排队论中的符号表示

排队论中的记号是 20 世纪 50 年代初由 D.G.Kendall 引入的，通常由 3~5 个字母组成，形式为：

$$A/B/C/n$$

其中  $A$  表示输入过程， $B$  代表服务时间， $C$  代表服务台数量， $n$  表示系统容量。如

(1)  $M/M/S/\infty$  表示输入过程是 Poisson 流，服务时间服从负指数分布，系统有  $S$  个服务台平行服务，系统容量为无穷大的等待制排队系统。

(2)  $M/G/S/\infty$  表示输入过程是 Poisson 流，服务时间服从一般概率分布，系统有  $S$  个服务台平行服务，系统容量为无穷大的等待制排队系统。

(3)  $D/M/S/K$  表示顾客相继到达时间间隔独立、服从定长分布，服务时间服从负指数分布，系统有  $S$  个服务台平行服务，系统容量为  $K$  个的混合制系统。

(4)  $M/M/S/S$  表示输入过程是 Poisson 流，服务时间服从负指数分布，系统有  $S$  个服务台平行服务，顾客到达后不等待的损失制系统。

(5)  $M/M/S/K/K$  表示输入过程是 Poisson 流，服务时间服从负指数分布，系统有  $S$  个服务台平行服务，系统容量和顾客容量都为  $K$  个的闭合制系统。

## 4.2.2 排队论中四种重要模型

### 1. 等待制模型 $M/M/S/\infty$

该模型中顾客到达规律服从参数为  $\lambda$  的 Poisson 分布，在  $[0,t]$  时间内到达的顾客数  $X(t)$  服从的分布为：

$$P\{X(t)=k\}=\frac{(\lambda t)^k \cdot e^{-\lambda t}}{k!} \quad (1)$$

其单位时间到达的顾客平均数为  $\lambda$ ， $[0,t]$  时间内到达的顾客平均数为  $\lambda t$ 。

顾客接受服务的时间服从负指数分布，单位时间服务的顾客平均数为  $\mu$ ，服务时间的分布为：

$$f(t)=\begin{cases} \mu e^{-\mu t} & t>0 \\ 0 & t=0 \end{cases} \quad (2)$$

每个顾客接受服务的平均时间为  $\frac{1}{\mu}$ 。





下面分别给出  $S=1$  和  $S>1$  的一些主要结果。

◆ 只有一个服务台的情形： $S=1$

可以计算出稳定状态下系统有  $n$  个顾客的概率：

$$p_n = (1 - \rho)\rho^n \quad n = 0, 1, 2, 3, \dots \quad (3)$$

其中  $\rho = \frac{\lambda}{\mu}$  称为系统的服务强度。

则系统没有顾客的概率为：

$$p_0 = 1 - \rho = 1 - \frac{\lambda}{\mu}$$

系统中顾客的平均队长为：

$$L_s = \sum_{n=0}^{\infty} n \cdot p_n = (1 - \rho) \sum_{n=0}^{\infty} n \cdot \rho^n = \frac{\rho}{1 - \rho} = \frac{\lambda}{\mu - \lambda} \quad (4)$$

系统中顾客的平均等待队长为：

$$L_q = \sum_{n=1}^{\infty} (n-1) \cdot p_n = (1 - \rho) \sum_{n=1}^{\infty} (n-1) \cdot \rho^n = \frac{\rho^2}{1 - \rho} = \frac{\lambda^2}{\mu(\mu - \lambda)} \quad (5)$$

系统中顾客的平均逗留时间为：

$$W_s = \frac{1}{\mu - \lambda} \quad (6)$$

系统中顾客的平均等待时间为：

$$W_q = \frac{1}{\mu - \lambda} - \frac{1}{\mu} = \frac{\lambda}{\mu(\mu - \lambda)} \quad (7)$$

从 (4) ~ (6) 式可以看出：

$$L_s = \lambda W_s, \quad L_q = \lambda W_q \quad (8)$$

$$\text{或} \quad W_s = \frac{L_s}{\lambda}, \quad W_q = \frac{L_q}{\lambda} \quad (9)$$

该公式称为 Little 公式。在其他排队论模型中依然适用。

Little 公式的直观意义：

$L_s = \lambda W_s$  表明排队系统的队长等于一个顾客平均逗留时间内到达的顾客数。

$L_q = \lambda W_q$  表明排队系统的等待队长等于一个顾客平均等待时间内到达的顾客数。

◆ 系统有多个服务台的情形： $S>1$

当系统中有  $S$  个服务台，系统服务能力为  $S\mu$ ，服务强度为  $\rho = \frac{\lambda}{S\mu}$ 。

系统中顾客的平均队长为：



$$L_s = S\rho + \frac{(S\rho)^s \rho}{S!(1-\rho)^2} \cdot p_0 \quad (10)$$

其中  $p_0 = \left[ \sum_{k=0}^{S-1} \frac{(S\rho)^k}{k!} + \frac{(S\rho)^s}{S!(1-\rho)} \right]^{-1}$ , 表示所有服务台都空闲的概率。

系统中顾客的逗留时间为:

$$W_s = \frac{L_s}{\lambda} \quad (11)$$

系统中顾客的平均等待时间为:

$$W_q = W_s - \frac{1}{\mu} \quad (12)$$

系统中顾客的平均等待队长为:

$$L_q = \lambda W_q \quad (13)$$

◆ Lingo 中的相关函数及相关参数计算公式

(1) 顾客等待概率的公式:

$$P_{\text{wait}} = @peb(\text{load}, S) \quad (14)$$

其中  $S$  为服务台个数,  $\text{load}$  为系统到达的载荷, 即  $\text{load} = \frac{\lambda}{\mu}$ 。

(2) 顾客的平均等待时间公式:

$$W_q = P_{\text{wait}} \frac{T}{S - \text{load}} \quad (15)$$

其中  $T$  为顾客接受服务的平均时间, 有  $T = \frac{1}{\mu}$ 。

当  $\text{load} > S$  时无意义, 表示当系统负荷超过服务台个数时, 排队系统达到不稳定状态, 队伍将越排越长。

$$(3) \text{ 系统中顾客的平均逗留时间 } W_s = W_q + \frac{1}{\mu} \quad (16)$$

$$(4) \text{ 系统中顾客的平均队长 } L_s = \lambda W_s \quad (17)$$

$$(5) \text{ 系统中顾客的平均等待队长 } L_q = \lambda W_q \quad (18)$$

**问题 1** 某机关接待室只有 1 名对外接待人员, 每天工作 10h, 来访人员和接待时间都是随机的。设来访人员按照 Poisson 流到达, 到达速率为  $\lambda = 8$  人/h, 接待人员的服务速率为  $\mu = 9$  人/h, 接待时间服从负指数分布。

(1) 计算来访人员的平均等待时间, 等候的平均人数。

(2) 若到达速率增大为  $\lambda = 20$  人/h, 每个接待人员的服务速率不变, 为使来访问人员

平均等待时间不超过半小时，最少应该配置几名接待人员。

解答：

(1) 该问题属于 M/M/1/∞ 排队模型。

$S=1$ ,  $\lambda=8$ ,  $\mu=9$  需要计算来访人员的平均等待时间  $W_q$ ，等候的平均人数  $L_q$ 。

Lingo 程序为：

```
model:
lp=8;
u=9;
T=1/u;
load=lp/u;
S=1;
Pwait=@PEB(load,S);!等待概率;
W_q=Pwait*T/(S-load);!平均等待时间;
L_q=lp*W_q;!顾客的平均等待队长;
end
```

计算结果：

来访人员的平均等待时间  $W_q = 0.89 \text{ h} = 53 \text{ min}$ ，等候的平均人数  $L_q = 7.1$  人。

(2) 属于 M/M/S/∞ 排队模型的优化问题。

求最小的  $S$ ，来访人员的平均等待时间  $W_q \leq 0.5$ 。

建立模型为：

$$\begin{aligned} \min S \\ \text{s.t.} \left\{ \begin{array}{l} P_{\text{wait}} = @peb(\text{load}, s) \\ \text{load} = \frac{\lambda}{\mu} \\ T = \frac{1}{\mu} \\ W_q = P_{\text{wait}} \frac{T}{S - \text{load}} \\ L_q = \lambda W_q \\ W_q \leq 0.5 \\ S \in N \end{array} \right. \end{aligned}$$

实现的 Lingo 程序为：

```
model:
min=S;
lp=20;
u=9; !服务率;
T=1/u;
load=lp/u;
Pwait=@PEB(load,S);!接待人员的等待概率;
W_q=Pwait*T/(S-load);!平均等待时间;
```





```

W_q<=0.5;
L_q=lp*W_q;!顾客的平均等待队长;
TT=W_q*60;
@gin(S);
end

```

计算结果为：最少需要接待人员  $S=3$  人。

来访人员等待概率为 0.55，排队等待平均时间为 4.7min，队长平均长度为 1.58 人。

## 2. 损失制模型 M/M/S/S

M/M/S/S 模型表示顾客到达人数服从 Poisson 分布，单位时间到达率为  $\lambda$ ，服务台服务时间服从负指数分布，单位时间服务平均人数为  $\mu$ 。当  $S$  个服务台被占用后，顾客自动离开，不再等待。这里我们给出 Lingo 中的有关函数及相关参数的计算公式。

(1) 系统损失概率

$$P_{\text{lost}} = @\text{pel}(\text{load}, S) \quad (19)$$

其中  $S$  为服务台服务台个数，load 为系统到达的载荷，即  $\text{load} = \frac{\lambda}{\mu}$ 。损失概率表示损失的顾客所占的比率。

(2) 单位时间内进入系统的平均顾客数

$$\lambda_e = \lambda(1 - P_{\text{lost}}) \quad (20)$$

(3) 系统中顾客的平均队长（系统在单位时间内占用服务台的均值）

$$L_s = \frac{\lambda_e}{\mu} \quad (21)$$

(4) 系统中顾客的平均逗留时间（服务时间）

$$W_s = \frac{1}{\mu} = T \quad (22)$$

(5) 系统服务台的效率

$$\eta = \frac{L_s}{S} \quad (23)$$

在损失制排队模型中，顾客平均等待时间  $W_q = 0$ ，平均等待队长  $L_q = 0$ ，因为没有顾客等待。

**问题 2** 某单位电话交换台有一部 300 门内线电话的总机，已知上班时间内有 30% 的内线分机平均每 30min 要一次外线电话，70% 的分机每隔 70min 时要一次外线电话。又知从外单位打来的电话的呼唤率平均 30s 一次，设与外线的平均通话时间为 3min，以上时间都服从负指数分布。如果要求外线电话接通率为 95% 以上，问电话交换台应设置多少外线？

**解：**该问题属于损失制排队的优化建模。



电话交换台的服务分为两部分，一类是内线打外线，一类是外线打内线。

内线打外线的服务强度（每 h 通话平均次数）

$$\lambda_1 = \left( \frac{60}{30} \times 30\% + \frac{60}{70} \times 70\% \right) \times 300 = 1.2 \times 300 = 360$$

外线打内线的服务强度

$$\lambda_2 = \frac{60}{0.5} = 120$$

总强度为  $\lambda = \lambda_1 + \lambda_2 = 360 + 120 = 480$

电话平均服务时间为  $T = \frac{3}{60} = 0.05$  h，服务率  $\mu = \frac{60}{3} = 20$  个

对该问题，目标是求最小的电话交换台数  $S$ ，使顾客（外线电话）损失率不超过 5%，即：

$$P_{lost} \leq 5\%$$

建立的优化模型为：

$$\begin{aligned} & \min S \\ & s.t. \begin{cases} P_{lost} = @pel(load, S) \\ load = \frac{\lambda}{\mu} \\ P_{lost} \leq 0.05 \\ \lambda_e = \lambda(1 - P_{lost}) \\ L_s = \frac{\lambda_e}{\mu} \\ \eta = \frac{L_s}{S} \\ S \in N \end{cases} \end{aligned}$$

Lingo 程序为：

```
model:
min=S;
lp=480; !每小时平均到达电话数;
u=20; !服务率;
load=lp/u;
Plost=@PEL(load,S); !损失率;
Plost<=0.05;
lpe=lp*(1-Plost);
L_s=lpe/u; !顾客的平均队长;
eta=L_s/S; !系统服务台的效率;
@gin(S);
end
```





计算结果为:

最小的电话交换台为  $S=30$ 。

电话损失率为  $P_{\text{lost}}=0.04$ ，实际进入系统的电话平均为  $\lambda_e=460.7$ ，平均队长  $L_s=23$ ，系统服务台的效率  $\eta=0.768$ 。

### 3. 混合制模型 M/M/S/K

混合制模型 M/M/S/K，表示顾客到达人数服从 Poisson 分布，单位时间到达率为  $\lambda$ ，服务台服务时间服从负指数分布，单位时间服务平均人数为  $\mu$ ，系统有  $S$  个服务台，系统对顾客的容量为  $K$ 。当  $K$  个位置被顾客占用时，新到的顾客自动离去。当系统中有空位置时，新到的顾客进入系统排队等候。

对混合制模型，Lingo 没有相关函数计算参数。需要自己编程计算。

#### (1) 混合制模型基本公式

设稳定状态下系统有  $i(i=0,1,2,\dots,K)$  个顾客的概率为  $p_i(i=0,1,2,\dots,K)$ 。 $p_0$  表示系统空闲的概率。因此

$$\sum_{i=0}^K p_i = 1 \quad p_i \geq 0, i=1,2,\dots,K \quad (24)$$

设  $\lambda_i(i=0,1,2,\dots,K)$  表示系统中有  $i$  个顾客时的输入强度， $\mu_i(i=0,1,2,\dots,K)$  表示系统中有  $i$  个顾客时的服务强度。在稳定状态下，可建立平衡方程

$$\begin{cases} \mu_1 p_1 = \lambda_0 p_0 \\ \lambda_0 p_0 + \mu_2 p_2 = (\lambda_1 + \mu_1) p_1 \\ \lambda_{i-1} p_{i-1} + \mu_{i+1} p_{i+1} = (\lambda_i + \mu_i) p_i \quad (i=1,2,\dots,K-1) \\ \lambda_{K-1} p_{K-1} = \mu_K \cdot p_K \end{cases} \quad (25)$$

对于混合制系统 M/M/S/K，有：

$$\lambda_i = \lambda \quad (i=0,1,2,\dots,K) \quad (26)$$

$$\mu_i = \begin{cases} i\mu & i \leq S \\ S\mu & i > S \end{cases} \quad (i=1,2,\dots,K) \quad (27)$$

#### (2) 混合制模型基本参数计算

由于当系统有  $K$  个人时，到达的顾客就会流失，因此有系统损失概率：

$$P_{\text{lost}} = P_K \quad (28)$$

单位时间内进入系统的平均顾客数

$$\lambda_e = \lambda(1 - P_{\text{lost}}) = \lambda(1 - P_K) \quad (29)$$

系统中顾客的平均队长

$$L_s = \sum_{i=0}^K i \cdot p_i \quad (30)$$

系统中顾客的平均等待队长

$$L_q = \sum_{i=S}^K (i-S) \cdot p_i = L_s - \frac{\lambda_e}{\mu} \quad (31)$$

系统中顾客的平均逗留时间

$$W_s = \frac{L_s}{\lambda_e} \quad (32)$$

系统中顾客的平均等待时间

$$W_q = W_s - \frac{1}{\mu} = W_s - T \quad (33)$$

**问题 3** 某理发店有 4 名理发师, 因场地所限, 店里最多容纳 12 名顾客, 假设来理发的顾客按 Poisson 分布到达, 平均到达率为 18 人/h, 理发时间服从负指数分布, 平均每人 12min。求该系统的各项指标。

**解:** 该模型是 M/M/4/12 混合制模型。\$S=4\$, \$K=12\$, \$\lambda=18\$, \$\mu=60/12=5\$。各项指标的计算采用上面的公式。

该程序的编制可采用 Lingo 实现。程序为:

```
!混合制排队论模型;
model:
sets:
state/1..12/:P;
endsets
lp=18; !顾客到达率;
u=5; !服务率;
S=4; !服务员人数;
K=12; !系统容量;
P0+@sum(state(i):P(i))=1; !概率和;
u*P(1)=lp*P0; !平衡点 0;
lp*P0+2*u*P(2)=(lp+u)*P(1); !平衡点 1;
@for(state(i)|i#GT#1#and#i#LT#S:lp*P(i-1)+(i+1)*u*P(i+1)=(lp+i*u)*P(i)); !平衡点 i[2,S-1];
@for(state(i)|i#GE#S#and#i#LT#K:lp*P(i-1)+S*u*P(i+1)=(lp+S*u)*P(i)); !平衡点 i[S,K-1];

lp*P(K-1)=S*u*P(K); !平衡点 K;

Plost=P(K); !损失率;
lpe=lp*(1-P(K)); !实际到达率;

L_s=@sum(state(i):i*P(i)); !平均队长;
L_q=L_s-lpe/u; !平均等待队长;
```





```

W_s=L_s/lpe; !平均逗留时间;
W_q=L_q/lpe; !平均等待时间;

end

```

计算结果为:

理发师空闲率为  $P_0=0.16$ , 损失顾客率  $0.049$ , 每小时实际进入理发店人数为  $\lambda_e=17.12$  人, 平均队长  $L_s=5.72$  人, 平均等待队长  $L_q=2.3$  人, 平均逗留时间  $W_s=0.334\text{h}$ , 平均等待时间  $W_q=0.134\text{h}$ 。

#### 4. 闭合制模型 M/M/S/K/K

M/M/S/K 模型表示系统有  $S$  个服务台, 顾客到达人数服从 Poisson 分布, 单位时间到达率为  $\lambda$ , 服务台服务时间服从负指数分布, 单位时间服务平均人数为  $\mu$ 。系统容量和潜在的顾客数都为  $K$ 。

基本参数计算:

(1) 平均队长 (Lingo 计算公式)

$$L_s = @pfs(\text{load}, S, K) \quad (34)$$

其中  $S$  为服务台服务台个数,  $\text{load}$  为系统到达的载荷, 这里  $\text{load} = K \cdot \frac{\mu}{\lambda}$ 。

(2) 单位时间平均进入系统的顾客数

$$\lambda_e = \lambda(K - L_s) \quad (35)$$

(3) 顾客处于正常情况的概率

$$P = \frac{K - L_s}{K} \quad (36)$$

(4) 系统中顾客的平均等待队长  $L_q$ , 平均逗留时间  $W_s$ , 平均等待时间  $W_q$  为:

$$L_q = L_s - \frac{\lambda_e}{\mu} \quad W_s = \frac{L_s}{\lambda_e} \quad W_q = \frac{L_q}{\lambda_e} \quad (37)$$

(5) 每个服务台的工作强度

$$P_{\text{work}} = \frac{\lambda_e}{S\mu} \quad (38)$$

**问题 4** 某工厂现有 30 台自动车床, 由 4 名工人负责维修管理。当机床需要加料、发生故障或刀具磨损时就自动停车, 等待工人照管。设平均每台机床两次停车时间间隔为  $1\text{h}$ , 停车时需要工人照管的平均时间为  $5\text{min}$ , 并服从负指数分布。求该排队系统的各项指标。

**解:** 该排队系统是闭合制排队模型 M/M/4/30/30。参数  $S=4$ ,  $K=30$ ,  $\lambda=1$ ,  $T=\frac{5}{60}=\frac{1}{12}$ ,  $\mu=12$ 。根据上面公式可计算出各项指标。



Lingo 程序为:

```
model:
lp=1;!每小时故障到达数;
u=12;!服务率;
K=30;!机器数;
S=4;!维修工人数;
load=K*lp/u;
L_s=@pfs(load,S,K);!等待队长;
lpe=lp*(K-L_s);!进入维修的平均机器数;
Prob=(K-L_s)/K;!机器工作概率;
L_q=L_s-lpe/u;!平均等待队长;
W_q=L_q/lpe;!平均等待时间;
W_s=L_s/lpe;!平均逗留时间;
Pwork=lpe/(S*u);!维修工人的工作强度;
end
```

计算结果为:

实际进入系统的机器平均为  $\lambda_e = 27.5$ , 平均队长  $L_s = 2.5$ , 平均等待队长  $L_q = 0.24$ , 平均逗留时间  $W_s = 0.09$ , 平均等待时间  $W_q = 0.087$ , 机器工作概率  $\text{Prob} = 0.92$ , 维修工人的工作强度  $P_{\text{work}} = 0.57$ 。

**问题 5** 某修理厂为设备检修服务。已知检验的设备(顾客)到达服从 Poisson 分布, 每天到达率  $\lambda = 42$  台, 当需要等待时, 每台造成的损失为 400 元。服务(检修)时间服从负指数分布, 平均每天服务率  $\mu = 20$  台。每设置一个检修人员每天的服务成本为 160 元。问设立几个检修人员才能使平均总费用最小?

**解:** 该排队系统为 M/M/S/ $\infty$  系统。系统参数中  $\lambda = 42$ ,  $\mu = 20$ 。

费用包括等待费用和人员费用。目标函数为:

$$\min z = 160S + 400L_s$$

其中  $S$  为维修人员数,  $L_s$  为平均队长。

优化模型为:

$$\begin{aligned} \min z &= 160S + 400L_s \\ \text{s.t.} \quad &\begin{cases} P_{\text{wait}} = @peb(\text{load}, S) \\ \text{load} = \frac{\lambda}{\mu} \\ W_q = P_{\text{wait}} \frac{T}{S - \text{load}} \\ L_q = \lambda W_q \\ L_s = L_q + \frac{\lambda}{\mu} \\ S \in N \end{cases} \end{aligned}$$





实现的 Lingo 程序为：

```
model:
min=160*S+400*L_s;
lp=42;!每天到达顾客数;
u=20;!每天服务人数;
load=lp/u;!载荷;
T=1/u;
Pwait=@PEB(load,S);!顾客的等待概率;
W_q=Pwait*T/(S-load);!平均等待时间;
L_q=lp*W_q;!顾客的平均等待队长;
L_s=L_q+lp/u;!平均队长;
s>=2;
@gin(S);
end
```

注：程序中加上  $S \geq 2$  是为了便于 Lingo 求解。因为当  $S=1$  时， $S - \text{load} < 0$  不符合要求，Lingo 无法求解，而实际中也需要满足  $S \geq 2$ ，故加上该条件。

计算结果为：

最小平均费用为 1568.17 元。

最优人员数  $S=4$ ，平均队长  $L_s=2.32$ 。

### 4.2.3 排队论的计算机模拟

排队论中的问题有的可以通过理论计算解决，有的则需要通过计算机模拟计算得到。当理论计算难以解决时，则可以考虑采用计算机模拟计算的方法来解决。

#### 问题 1 收款台服务问题

考虑一个收款台的排队系统。某商店只有一个收款台，顾客到达收款台的时间间隔服从平均时间为 10s 的负指数分布。负指数分布为：

$$f(x) = \begin{cases} \frac{1}{\lambda} e^{-\frac{x}{\lambda}} & x > 0 \\ 0 & x \leq 0 \end{cases}$$

每个顾客的服务时间服从均值为 6.5s，标准差为 1.2s 的正态分布。利用计算机仿真计算顾客在收款台的平均逗留时间，系统的服务强度（服务占所有时间之比）。

分析：该问题中顾客服务时间服从正态分布，不再是负指数分布，不能直接采用前面的模型计算，因此我们可以考虑采用计算机模拟计算得到需要的结果。

该问题可以从开始时刻计，当有人到达时产生一个事件，离开时产生一个事件。当有人到达时，记录其开始接受服务时刻和离开服务台的时刻，从而可以计算出每个人在系统逗留的时间，以及每个人在系统接受服务的时间，从而统计出每个人在收款台的平均逗留时间和系统的服务强度。



这里我们可以依次考虑每一个人，考察其到达时刻，开始接受服务时刻和离开时刻，使仿真变得更方便。

设第 $i$ 个人到达时间为 $a_i$ ，开始接受服务的时间为 $b_i$ ，离开时间为 $c_i$ ，并设总共考虑 $n$ 个人。

程序首先产生服从均值为10s的负指数分布序列 $\{dt(n)\}$ ，每个人接受服务时间服从正态分布 $N(6.5, 1.2^2)$ 的序列 $\{st(n)\}$ 。便于为后面计算方便。

则每个人的到达时刻可以采用下式计算

$$a_1 = 0, \quad a_i = a_{i-1} + dt_{i-1} \quad i = 2, 3, \dots, n$$

第一个人开始接受服务时刻 $b_1 = 0$

第一个人服务时间 $c_1 = st_1$

第 $i$ 个人开始接受服务的时刻为

$$b_i = \begin{cases} a_i & a_i > c_{i-1} \\ c_{i-1} & a_i \leq c_{i-1} \end{cases} \quad i = 2, 3, \dots, n$$

上式的意义是当后一个人到达时刻比前一个人离开时刻晚，则其开始接受服务时间就是其到达时间；当后一个人到达时刻比前一个人离开时刻早，则其开始接受服务时间就是前一个人的离开时刻。

第 $i$ 个人离开时刻为

$$c_i = b_i + st_i \quad i = 2, 3, \dots, n$$

根据上面的递推关系式就可以计算出每个人到达时刻、开始接受服务时刻和离开时刻。每个人在系统逗留时间为

$$wt_i = c_i - a_i \quad i = 1, 2, \dots, n$$

到第 $n$ 个人离开的时刻为 $T = c_n$ ，则系统工作的强度及工作时间占总时间比值为

$$p = \sum_{i=1}^n st_i / T$$

以下为MatLab模拟计算程序：

```
n=10 000; %模拟顾客数
dt=exprnd(10,1,n); %到达时间间隔
st=normrnd(6.5,1.2,1,n); %服务台服务时间
%st=exprnd(2.5,1,n); %服务台服务时间
a=zeros(1,n); %每个人到达时间
b=zeros(1,n); %每个人开始接受服务时间
c=zeros(1,n); %每个人离开时间

a(1)=0;
```





```

for i=2:n
    a(i)=a(i-1)+dt(i-1);%第 i 个人到达时间
end

b(1)=0;%第 1 个人开始服务时间为到达时间
c(1)=b(1)+st(1); %第 1 个人离开时间为服务时间

for i=2:n

    if(a(i)<=c(i-1)) b(i)=c(i-1);%如果第 i 个人到达时间比前一个人离开时间早,则其开始
    服务时间为前一人离开时间
    else b(i)=a(i); %如果第 i 个人到达时间比前一个人离开时间晚,则其开始服务时间为到达
    时间.
    end

    c(i)=b(i)+st(i); %第 i 个人离开时间为其开始服务时间+接受服务时间

end

cost=zeros(1,n); %记录每个人在系统逗留时间
for i=1:n
    cost(i)=c(i)-a(i); %第 i 个人在系统逗留时间
end

T=c(n);%总时间
p=sum(st)/T; %服务率
avert=sum(cost)/n; %每个人系统平均逗留时间
fprintf('顾客平均逗留时间;%6.2f 秒\n',avert);
fprintf('系统工作强度;%6.3f\n',p);

```

某次仿真结果为:

顾客平均逗留时间 13.32s

系统工作强度 0.659

## 问题 2 卸货问题

某码头有一卸货场, lunchao 轮船一般夜间到达, 白天卸货。每天只能卸货 4 艘船, 若一天内到达数超过 4 艘, 那么推迟到第二天卸货。根据过去经验, 码头每天船达到数服从表 4-11 所列的概率分布。求每天推迟卸货的平均船数。

表 4-11

到达船数	0	1	2	3	4	5	6	7	$\geq 8$
概 率	0.05	0.1	0.1	0.25	0.2	0.15	0.1	0.05	0

解答: 该问题可以看作单服务台的排队系统。到达时间不服从负指数分布, 服从的是给定的离散分布, 服务时间也不服从负指数分布, 是定长时间服务。不能直接利用理论公式求解, 可采用计算机模拟器求解。

## (1) 随机到达船数的产生

首先我们需要产生每天随机到达的船数,该随机数服从离散分布,可以先产生一个 0~1 之间的均匀随机数,其落在不同区间则寿命取不同值,具体见表 4-12 所列。程序实现函数见 BoatNumber.m,每调用一次该函数,则返回一个服从该分布的船数。

表 4-12

到达船数	均匀随机数区间
0	(0,0.05)
1	(0.05,0.15)
2	(0.15,0.25)
3	(0.25,0.5)
4	(0.5,0.7)
5	(0.7,0.85)
6	(0.85,0.95)
7	(0.95,1)

## (2) 计算机仿真分析:

设第  $i$  天到达的船数为  $x_i$  艘,需要卸货的船数为  $a_i$  艘,实际卸货的船数为  $b_i$  艘,推迟卸货的船数为  $d_i$  艘。

设总共模拟  $n$  天,首先模拟  $n$  天的到达船数  $x_1, x_2, \dots, x_n$ 。根据该问题要求,各个量之间有如下关系:

初始第一天,推迟卸货的船数为  $d_1 = 0$  艘。

第  $i$  天需要卸货的船数  $a_i$  满足:  $a_i = x_i + d_i \quad i = 1, 2, \dots, n$

第  $i$  天实际卸货的船数  $b_i$  满足:  $b_i = \begin{cases} 4 & a_i > 4 \\ a_i & a_i \leq 4 \end{cases} \quad i = 1, 2, \dots, n$

第  $i$  天推迟卸货的船数  $d_i$  满足:  $d_i = a_i - b_i \quad i = 2, \dots, n$

则总共推迟卸货的船数为

$$\text{Total} = \sum_{i=1}^n d_i$$

每天推迟卸货的平均船数  $Aver = \text{Total} / n$

下面是 MatLab 实现程序

## ① 产生元件寿命的随机值函数 BoatNumber.m

function X=BoatNumber

```
%产生一个到达船数的随机数
```

```
Boat=0:7; %到达船数取值范围
```

```
%到达船数概率分布
```

```
Prob=[0.05,0.1,0.1,0.25,0.20,0.15,0.1,0.05];
```

```
n=length(Prob);
```

```
Qu=zeros(1,n+1);
```





```

Qu(1)=0;
for i=1:n
    Qu(i+1)=Qu(i)+Prob(i); %产生概率区间
end
Qu(n+1)=1.01; %将最后一个数值超过 1,便于后面的随机数 r 取到 1

%产生一次到达船数
r=rand(1); %产生一个[0,1]随机变量
for i=1:n
    if(r>=Qu(i)&&r<Qu(i+1)) X=Boat(i); %获得到达船数
end
end
return

```

## ② 模拟计算的主程序 Boat.m

```

n=10000; %模拟总天数
x=zeros(n,1); %存储每天到达船数
a=zeros(n,1); %存储每天需要卸货的船数
b=zeros(n,1); %存储每天实际卸货的船数
d=zeros(n,1); %存储每天推迟卸货的船数

for i=1:n
    x(i)=BoatNumber; %模拟 n 天到达船数
end

for i=1:n
    a(i)=x(i)+d(i); %计算每天需要卸货的船数
    if a(i)>4 b(i)=4; %计算每天实际卸货的船数
    else b(i)=a(i);
    end
    if i>=2 d(i)=a(i)-b(i); end %计算每天推迟卸货的船数
end

Total=sum(d); %计算总共推迟卸货船数
Aver=Total/n; %计算每天推迟卸货的平均船数
fprintf('每天推迟卸货的平均船数%.2f\n',Aver);

```

某次模拟结果为：

每天推迟卸货的平均船数 0.51

多次模拟计算，每天推迟卸货的平均船数在 0.5 艘左右。

## 问题 3 眼科病床的合理安排（CUMCM2009B）

医院就医排队是大家都非常熟悉的现象，它以这样或那样的形式出现在我们面前，例如，患者到门诊就诊、到收费处划价、到药房取药、到注射室打针、等待住院等，往往需要排队等待接受某种服务。



我们考虑某医院眼科病床的合理安排的数学建模问题。

该医院眼科门诊每天开放,住院部共有病床 79 张。该医院眼科手术主要分四大类:白内障、视网膜疾病、青光眼和外伤。附录中给出了 2008 年 7 月 13 日至 2008 年 9 月 11 日这段时间里各类病人的情况。

白内障手术较简单,而且没有急症。目前该院是每周一、三做白内障手术,此类病人的术前准备时间只需 1、2 天。做两只眼的病人比做一只眼的要多一些,大约占到 60%。如果要做双眼是周一先做一只,周三再做另一只。

外伤疾病通常属于急症,病床有空时立即安排住院,住院后第二天便会安排手术。

其他眼科疾病比较复杂,有各种不同情况,但大致住院以后 2~3 天内就可以接受手术,主要是术后的观察时间较长。这类疾病手术时间可根据需要安排,一般不安排在周一、周三。由于急症数量较少,建模时这些眼科疾病可不考虑急症。

该医院眼科手术条件比较充分,在考虑病床安排时可不考虑手术条件的限制,但考虑到手术医生的安排问题,通常情况下白内障手术与其他眼科手术(急症除外)不安排在同一天做。当前该住院部对全体非急症病人是按照 FCFS (First come, First serve) 规则安排住院,但等待住院病人队列却越来越长,医院方面希望你们能通过数学建模来帮助解决该住院部的病床合理安排问题,以提高对医院资源的有效利用。

问题一:试分析确定合理的评价指标体系,用以评价该问题的病床安排模型的优劣。

问题二:试就该住院部当前的情况,建立合理的病床安排模型,以根据已知的第二天拟出院病人数来确定第二天应该安排哪些病人住院。并对你们的模型利用问题一中的指标体系作出评价。

问题三:作为病人,自然希望尽早知道自己大约何时能住院。能否根据当时住院病人及等待住院病人的统计情况,在病人门诊时即告知其大致入住时间区间。

问题四:若该住院部周六、周日不安排手术,请你们重新回答问题二,医院的手术时间安排是否应作出相应调整?

问题五:有人从便于管理的角度提出建议,在一般情形下,医院病床安排可采取使各类病人占用病床的比例大致固定的方案,试就此方案,建立使得所有病人在系统内的平均逗留时间(含等待入院及住院时间)最短的病床比例分配模型。

附件数据可参见网址: <http://www.mathor.com/thread-11976-1-1.html>。

该问题中问题五可用排队论模型求解。由于其他问题放在一起考虑太繁琐,这里只考虑用排队论方法求解第五问。

我们每张病床看作一个服务台,病人看作顾客,由此可采用排队论模型求解。

将服务台看作 5 种类型,每种类型对应一种疾病的病床。则每种类型的服务台数量就是我们需要优化的。对每种类型的服务台都可以看作  $M/M/S/\infty$  模型,然后将 5 种类型的服务台当并联处理。

由排队论知识可知,对  $M/M/S/\infty$  模型,当第  $i$  种服务台有  $y_i$  台,服务能力为  $y_i\mu_i$ ,服



务强度为  $\rho_i = \frac{\lambda_i}{y_i \mu_i} = \frac{\lambda_i T_i}{y_i}$ 。这里  $\lambda_i$  表示第  $i$  种病人平均每天的到达人数,  $\mu_i$  表示第  $i$  种病床每天服务的病人数,  $T_i$  代表第  $i$  种病人的平均住院时间。

第  $i$  种病人的平均人数 (含等待及住院人数)

$$LS_i = y_i \rho_i + \frac{(y_i \rho_i)^{y_i} \rho_i}{y_i! (1 - \rho_i)^2} \cdot p_{i0} \quad (i = 1, 2, 3, 4, 5)$$

其中  $p_{i0} = \frac{1}{\sum_{k=0}^{y_i-1} \frac{(y_i \rho_i)^k}{k!} + \frac{(y_i \rho_i)^{y_i}}{y_i! (1 - \rho_i)}}$ , 表示第  $i$  种病人的所有病床都空闲的概率。

由于对附录中数据进行检验, 发现每种病人手术后住院时间并不服从负指数分布, 因此应考虑为一般分布的模型  $M/G/S/\infty$ , 其中  $G$  表示一般分布模型, 只要求每个顾客接受服务的时间独立同分布就可以。此时排队长度有经验公式

$$GS_i = LS_i \cdot \frac{1 + v^2}{2} \quad (i = 1, 2, 3, 4, 5)$$

其中  $LS_i$  为根据  $M/M/S/\infty$  模型计算的排队长度。 $v$  为系统服务时间的偏离系数,  $v = \sigma\mu$ ,  $\sigma$  为服务时间的标准差。则

$$GS_i = LS_i \cdot \frac{1 + (\sigma\mu)^2}{2} \quad (i = 1, 2, 3, 4, 5)$$

第  $i$  种病人的排队 (等待住院) 时间为

$$WS_i = \frac{LS_i}{\lambda_i}$$

我们目标函数为所有病人的平均逗留时间最小。考虑 5 种病人到达的人数分布不同, 因此需要考虑权重。设第  $i$  种病的权重为  $w_i$  ( $i = 1, 2, 3, 4, 5$ )。由于每种病的权重跟其到达人数有关, 因此我们取  $w_i = \frac{\lambda_i}{\sum_{i=1}^5 \lambda_i}$  ( $i = 1, 2, 3, 4, 5$ )。这样目标函数为病人平均时等待住院时间

最小。即

$$\min Z = \sum_{i=1}^5 WS_i \cdot w_i$$

同时我们设决策变量为第  $i$  种病床  $y_i$  张。所有的病床数满足

$$\sum_{i=1}^5 y_i = 79$$

因此我们得到总的非线性整数规划模型为

$$\min Z = \sum_{i=1}^5 WS_i \cdot w_i$$



$$s.t. \left\{ \begin{array}{l} LS_i = y_i \rho_i + \frac{(y_i \rho_i)^{y_i} \rho_i}{y_i! (1 - \rho_i)^2} \cdot p_{i0} \quad (i=1,2,3,4,5) \\ GS_i = LS_i \cdot \frac{1 + (\sigma\mu)^2}{2} \quad (i=1,2,3,4,5) \\ p_{i0} = \frac{1}{\sum_{k=0}^{y_i-1} \frac{(y_i \rho_i)^k}{k!} + \frac{(y_i \rho_i)^{y_i}}{y_i! (1 - \rho_i)}} \\ \rho_i = \frac{\lambda_i T_i}{y_i} \\ WS_i = \frac{GS_i}{\lambda_i} \\ \sum_{i=1}^5 y_i = 79 \\ y_i \text{取整} \end{array} \right.$$

该模型不容易求解，我们寻求近似解答。

我们的基本思想是当各类病人构成的排队系统服务系统强度相同时，系统服务效率最高。

设5类病人：视网膜疾病，白内障（单眼），白内障（双眼），外伤，青光眼的到达率为 $\lambda_i$  ( $i=1,2,3,4,5$ )，平均住院时间为 $T_i$  ( $i=1,2,3,4,5$ )。

则第 $i$ 种病床构成的服务系统的强度为

$$\rho_i = \frac{\lambda_i}{y_i \cdot \mu_i} = \frac{\lambda_i T_i}{y_i} \quad (i=1,2,3,4,5)$$

总床位满足 $\sum_{i=1}^5 y_i = 79$ 。

令5种病床构成的服务系统强度相等，则有

$$\rho_i = \frac{\sum_{j=1}^5 \rho_j \cdot y_j}{79} = \frac{\sum_{j=1}^5 \lambda_j T_j}{79}$$

则

$$y_i = \frac{\lambda_i T_i}{\rho_i} = 79 \cdot \frac{\lambda_i T_i}{\sum_{j=1}^5 \lambda_j T_j} \quad (i=1,2,3,4,5)$$

对5种病人的数据进行统计，可以得到到达率为 $\lambda$ 分别为，2.7869, 1.6393, 2.1803, 1.0492, 1.0328；平均住院时间 $T$ 分别 12.545, 5.236, 8.561, 7.036, 10.487，（单位：天），容易按照上式计算得到病床数的分配如表4-13所列。





表 4-13

类 型	视网膜病	白内障（单眼）	白内障（双眼）	外 伤	青光眼
病床数	33	9	18	8	11
百分比	42.77%	11.4%	22.78%	10.13%	13.92%

## 4.3 元胞自动机简介及其在数学建模中的应用

### 4.3.1 引言

元胞自动机（Cellular Automata, CA）<sup>[1]</sup>，又称细胞自动机，最早由 John von Neumann 在 1950 年为模拟生物细胞的自我复制而提出的。当时 John von Neumann 假想有一台机器漂浮在池塘上面，这个池塘里有很多机器的零部件，这个机器是一台通用的建造器：只要给出它自身的描述，这台机器就能在池塘里寻找合适的零件再创造出自己。但这个模型仍然是以具体物质原材料的吸收为前提，后来在 Stanislaw M.Ulam 的建议下，他从细胞视角思考这个问题，于是元胞自动机就诞生了。作为描述自然界复杂现象的简化数学模型，元胞自动机可以表现出极其复杂的形态，因此常用于复杂系统的建模。20 世纪 50 年代，John von Neuman 和 Stanislaw M.Ulam 提出元胞自动机后，并未受到学术界重视。直到 1970 年，剑桥大学的 John Horton Conway 设计了一个电脑游戏——“生命游戏（Game of Life）”<sup>[2]</sup>后才吸引了科学家们的注意。此后，Stephen Wolfram 对初等元胞机 256 种规则<sup>[3]</sup>所产生的模型进行了深入研究，并用熵来描述其演化行为，将细胞自动机分为平稳型，周期型，混沌型和复杂型<sup>[1,4]</sup>。

元胞自动机是复杂系统研究的一个典型方法，特别适合用于空间复杂系统的时空动态模拟研究。不同于一般的动力学模型，元胞自动机不是由严格定义的物理方程或函数确定，而是用一系列模型构造的规则构成。凡是满足这些规则的模型都可以算是元胞自动机模型。因此，元胞自动机是一类模型的总称，或者说是一个方法框架。Stephen Wolfram 给出了元胞自动机需要满足的五个最基本要素：

（1）包含元胞（Cell）的规则的网络（Lattice Grid）。空间被离散成网格，元胞分布在网格中的各个格子中。

（2）离散的时间步长。元胞状态的改变是定义在两个时间步之间的，同一个时间步内，元胞的状态是固定的。

（3）每个元胞的状态数是有限的。也就是说任何元胞只能在固定的几种状态间切换。

（4）所有元胞状态的改变更新都遵循同样的规则。也就是说系统中的每一个元胞都是一样的，没有享受特殊待遇的元胞。

（5）元胞自动机的规则定义在局部。也就是说元胞的状态改变只取决于它周围的元胞的状态。

在元胞自动机模型中，散布在规则格网中的每一元胞取有限的离散状态，遵循同样的



作用规则，依据确定的局部规则作同步更新。大量元胞通过简单的相互作用而构成动态系统的演化。其特点是时间，空间，状态都离散，每个变量只取有限多个状态，且其状态改变的规则在时间和空间上都是局部的。

随着计算机的迅速发展，元胞自动机这种非常适合在计算机上实现的模型越来越被受到重视。目前，元胞自动机已经被广泛应用于社会学，经济学，生态学，图形学，数学，物理学，化学，地理，环境和军事学等各个领域，而这些领域都是潜在的数学建模竞赛的出题范围。这里仅列举出元胞自动机在以下三个领域的应用。

### 1. 社会学

元胞自动机经常用于研究个人行为的社会性问题的流行现象。例如人口的迁移，公共场所内人员的疏散，流行病的传播。这类社会学问题也是潜在的数学建模竞赛问题，比如1999年美国大学生数学建模B题“非法”聚会问题<sup>[5]</sup>就涉及公共场所内人员的疏散问题。

### 2. 图形学

元胞自动机以其特有的结构的简单性，内在的并行性以及复杂计算的能力成为密码学中研究的热点方向之一。例如被用于图像的加密。现在的数学建模问题也有很多涉及图形学的，比如2013年全国大学生数学建模B题纸片拼接问题<sup>[6]</sup>。

### 3. 物理学

在物理学中，元胞自动机已成功地应用于流体、磁场、电场、热传导等的模拟。例如格子气自动机。涉及物理学的数学建模问题也非常多，比如2013年MCM比赛A题布朗尼烤盘的最优设计问题<sup>[7]</sup>就涉及热传导。

元胞自动机是数学建模竞赛中最常用的方法之一。在近十年来（2000年后）的美国大学生数学建模竞赛（MCM）和中国大学生数学建模竞赛（CUMCM）中，几乎每两年就会出现适合应用元胞自动方法的赛题。表4-14为近十年来美国大学生数学建模竞赛中用到元胞自动机的特等奖论文不完全统计。

表 4-14

年份题号	题目	特等奖论文个数
2001MCM-B	逃避飓风	1
2003MCM-B	Gamma 刀治疗方案	1
2005MCM-B	收费亭的最优数量	3
2007MCM-B	飞机座位方案	4
2009MCM-A	交通环岛的设计	2
2012MCM-B	沿着“大长河”露营	1
2014MCM-A	交通右行规则	6

从表4-14中不难发现，用到元胞自动机的大多数模问题都与交通相关或类似。交通问题是数学建模竞赛中很常见的问题，值得一提的是2013年中国大学生数学建模竞赛A题“车道被占用对城市道路通行能力的影响”<sup>[6]</sup>也是典型的交通问题，获得一等奖的论文中





也有很多是用元胞自动机做的<sup>[8]</sup>。本书的编写者之一，北京理工大学的周登岳同学及其队友就应用元胞自动机模型来解决这一问题，获得了 2014 年全国大学生数模竞赛一等奖兼 IBM SPASS 奖<sup>[9]</sup>。在本章的第 3 小节中将具体介绍典型的交通流元胞自动机模型和编程实现方法。

### 4.3.2 方法介绍

在本章的第 1 节中已经给出了元胞自动机所需要满足的要素。应用元胞自动机模型来模拟动力学系统，首先需要给出这些要素的定义。在抽象出元胞自动机的各要素前，本节先介绍两个经典的元胞自动机，以使读者对元胞自动机先有个感性认识。

第一个要介绍的是在第 1 节中提到的“生命游戏”<sup>[2]</sup>。“生命游戏”是一个典型的二维元胞自动机，它是 Conway 在 20 世纪 60 年代末设计的，它包括一个二维正方形网格世界，这个世界中的每个方格居分布着一个活着的或死了的细胞。一个细胞在下一个时刻生死取决于相邻 8 个方格中活着的或死了的细胞的数量。如果相邻方格活着的细胞数量过多，这个细胞会因为资源匮乏而在下一个时刻死去；相反，如果周围活细胞过少，这个细胞会因太孤单而死去。生命游戏中元胞只具有“生”和“死”两种状态（在计算机中一般用 1 和 0 来分别表示生和死两种状态）；所有元胞在离散的时间步长上更新状态。一个元胞的生死由该时刻本身的状态和周围 8 个邻居的状态决定，具体规则如下：

(1) 当前元胞状态为“死”时，在周围有 3 个状态为“生”的元胞时，该元胞状态变成“活”。这一规则用来模拟生命的繁殖。

(2) 当前元胞状态为“活”时，在周围低于 2 个（不包含 2 个）状态为“活”的元胞时，该元胞状态变成“死”。这一规则用来模拟生命数量稀少时，个体也难以存活。

(3) 当前元胞状态为“活”时，在周围有 2 个或 3 个状态为“生”的元胞时，该元胞状态继续保持为“生”。

(4) 当前元胞状态为“活”时，在周围有 3 个（不包含 3 个）以上的状态为“活”的元胞时，该元胞状态变成“死”。这一规则用来模拟生命数量过多，竞争导致个体消亡。

以上四条规则可以简化为以下两条规则：

- 死亡规则：当前时刻，如果某元胞状态为“生”，且 8 个邻居元胞中有 2 个或 3 个状态为“生”，则下一时刻该元胞继续保持为“生”，否则“死”，图 4-15 所示是这一规则的示意图；

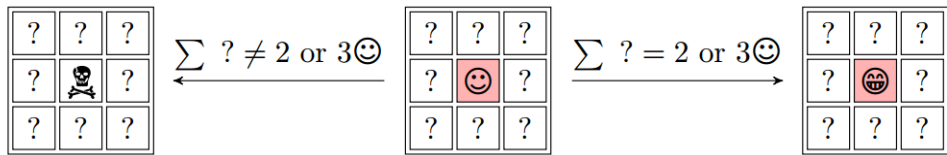


图 4-15

- 繁殖规则：当前时刻，如果某元胞状态为“死”，且 8 个邻居元胞中正好有 3 个状

态为“生”，则下一时刻该元胞复活为“生”，否则保持为“死”，图 4-16 所示是这一规则的示意图。



图 4-16

实际中，玩家可以设定周围活细胞的数目怎样时才适宜该细胞的生存。如果这个数目设定过低，世界中的大部分细胞会因为找不到太多的活的邻居而死去，直到整个世界都没有生命；如果这个数目设定过高，世界中又会被生命充满而没有什么变化。通常这个数目一般选取 2 或者 3，这样整个生命世界才不至于太过荒凉或拥挤，而是一种动态的平衡。由于设置的初始状态和迭代次数不同，游戏过程中会得到各种令人叹服的优美图案：杂乱无序的细胞会逐渐演化出各种精致，有形的结构；这些结构往往有很好的对称性，而且每一代都在变化形状；一些形状已经锁定，不会逐代变化；有时，一些已经成形的结构会因为一些无序细胞的“入侵”而被破坏。但是形状和秩序经常能从杂乱中产生出来。图 4-17 所示为游戏过程中得到的几种模式示意图。

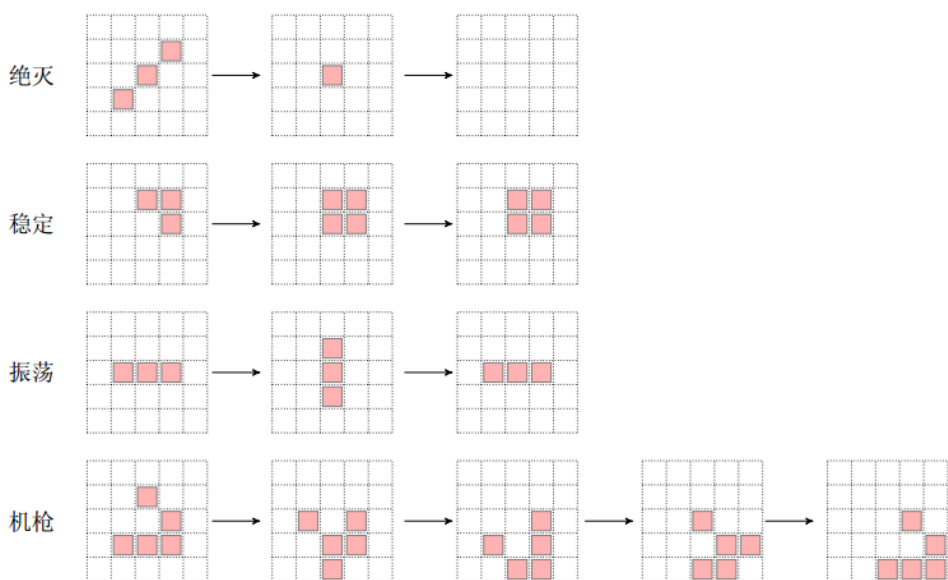


图 4-17

Wolfram 认为生命游戏的缺点是只研究了一种规则，他要尽可能系统地研究各种规则元胞自动机的行为。先从一维状态链做起，例如演化规则的半径为  $r=1$  的情况，包括自身共 3 个邻居。若每个位置可有 2 个状态，则这 3 个邻居可以取  $2^3=8$  种排列，对每一种排列，下一个时刻可以取 2 个不同的值，因此一共有  $2^{2^3}=256$  个不同的元胞自动机。本节要介绍

的第二个例子就是这 256 种一维元胞自动机中的第 184 种, 通常称为第 184 号规则, 如图 4-18 所示。184 号元胞自动机中也是最基本的交通流元胞自动机模型。

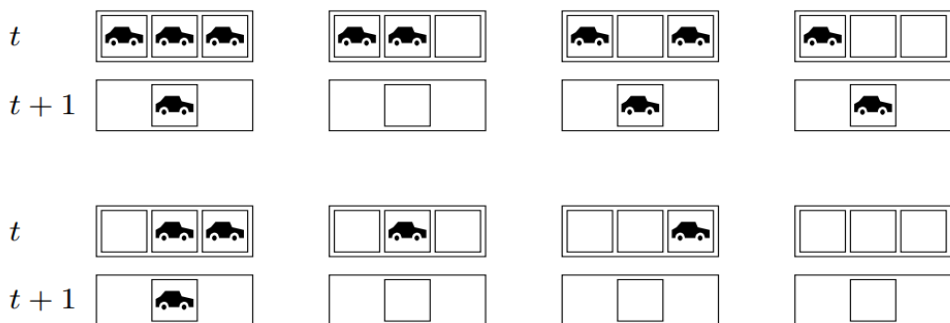


图 4-18

图 4-18 中的 184 号规则给出了相邻三个元胞的所有 8 种可能状态在下一时刻中间位置元胞的状态。在 184 号规则中, 道路被划分为等间距的格子, 每一个格子表示一个元胞。任一元胞只有两种状态, 或者被一辆占据, 或者为空。所有车辆都向同一方向行进(向右), 在某一时间步内, 如果第  $n$  辆车前方元胞为空, 则该车辆向前行驶一格, 否则保持原地不动。在 184 号规则下, 只要给定道路初始时刻的状态, 可以得到之后任何不同时刻道路的状态。如图 4-19 所示是一条长度为 20 的道路, 道路中有 5 辆车, 根据 184 号规则得到的前 10 个时间步的元胞自动机状态。每一行代表一维元胞自动机在某时刻的状态, 从上到下时间逐行增加一步。需要注意的是, 由于第一个元胞和最后一个元胞分别缺少左邻居和右邻居, 一种简单的做法是以最后一个元胞作为第一个元胞的左邻居, 同时也以第一个元胞作为最后一个元胞的右邻居。这样操作后, 相当于这一条路段是首尾相接的圆环, 从最右端驶出路段的车辆又从最左端驶入。这种边界的处理方式称为周期性边界条件。类似于图 4-19 所示的图, 一般称为时空图(横向表示空间, 纵向表示时间)。

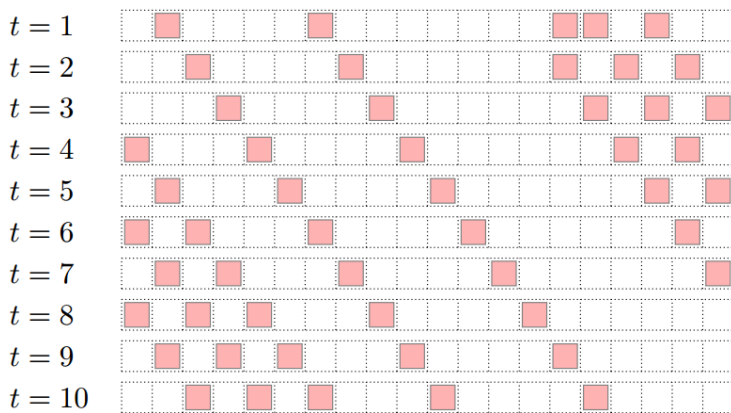


图 4-19

通过以上生命游戏元胞自动机和 184 号规则交通元胞自动机的介绍, 相信读者对元胞



自动机和元胞自动机的构成已经有了一定的感性认识。一个标准的元胞自动机 ( $A$ ) 由元胞, 元胞状态, 邻域状态及状态更新规则构成, 用数学式子表示为

$$A=(L, d, S, N, f)$$

其中  $L$  为元胞空间;  $d$  为元胞自动机内元胞空间的维数;  $S$  是元胞有限的, 离散的状态集合;  $N$  为某个邻域内所有元胞的集合;  $f$  为局部映射或局部规则。一个元胞通常在一个时刻只有取自一个有限集合的一种状态, 例如  $\{0,1\}$ 。元胞状态可以代表个体的性质, 特征, 行为等。在空间上与元胞相邻的细胞称为邻居, 所有邻居组成邻域。在计算机中, 元胞所在的网格一般具有边界, 处于边界上的元胞, 它周围的邻居元胞数量小于普通的元胞。因此对于边界上的元胞需要给出边界条件, 使所有元胞都具有相同数量的邻居。常见的边界条件有周期型, 反射型和定值型。接下来本节将具体介绍元胞自动机的各个构成要素:

### 1. 元胞

元胞是元胞自动机模型中最基本的单元, 就如同每一个细胞是生命机体组织中的一个最基本的单元。因此元胞自动机也叫细胞自动机。作为规则网格中的一个格子, 元胞可以有很多形状。元胞根据它周围的邻居元胞的状态, 按照规则来改变状态。状态可以是  $\{0,1\}$  的二进制形式。或是  $\{s_0, s_1, \dots, s_n\}$  整数形式的离散集, 严格意义上, 元胞自动机的元胞只能有一个状态变量。但在实际应用中, 往往将其进行了扩展, 每个元胞可以拥有多个状态变量。

### 2. 网格

所有元胞都是放在网格中的, 就像细胞是放在生物体上一样。从维度上分, 网格主要有一维, 二维, 三维及更高维。最简单的情况就是一维网格, 一维网格就是一系列的元胞排成一条线。在模拟单条车道的交通流时就是用的一维网格。最常用的是二维网格, 所有元胞排成在一个面内。图 4-20~图 4-22 所示是三种常见的二维网格。

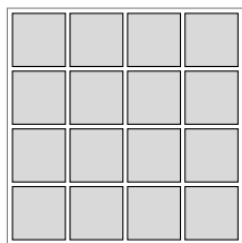


图 4-20

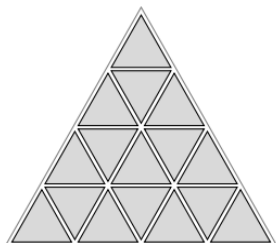


图 4-21

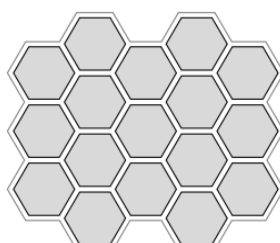


图 4-22

这三种网格各有优缺点: 三角形网格 (图 4-21) 的优点是拥有相对较少的邻居数目, 这在某些时候很有用; 其缺点是在计算机的表达与显示不方便, 需要转换为正方形网格。正方形网格 (图 4-20) 的优点是直观而简单, 而且特别适合于在现有计算机环境下进行表



达显示；其缺点是不能较好地模拟各向同性的现象，例如格子气模型中的 HPP 模型。六边形网格（图 4-22）的优点是能较好地模拟各向同性的现象，因此，模型能更加自然而真实，如格子气模型中的 FHP 模型，其缺点同三角网格一样，在表达显示上较为困难和复杂。在数学建模中，最常用的二维网格是正方形网格，一是正方形网格本身简单，并且一般情况的二维问题都适用，二是正方形网格可以用矩阵或数组直接表示，方便程序的实现。比如模拟多条车道时，车可能会换道，这时一维网格不能满足要求，需要用二维正方形网格，而三维及三维以上的网格较为复杂，在数学建模中很少用到。2003 年 MCM 的 Gamma 刀问题中，有一篇特等奖论文<sup>[10]</sup>，用元胞自动机来模拟肿瘤生长来生成一个随机的肿瘤。

### 3. 邻居

以上的元胞及元胞的网格空间只表示了系统的静态成分，为“动态”引入系统，必须加入演化规则。在元胞自动机中，这些规则是定义在空间局部范围内的，即一个元胞下一时刻的状态决定于本身状态和它的邻居元胞的状态。因而，在指定规则之前，必须定义一定的邻居规则，明确哪些元胞属于该元胞的邻居。在一维元胞自动机中，通常以半径来确定邻居，距离一个半径内的所有元胞均被认为是该元胞的邻居。二维元胞自动机的邻居定义较为复杂。在二维的方型网格中，常用的邻居有三种，如图 4-23~图 4-25 所示，分别是以周围 4 个（图 4-23），8 个（图 4-24）和 24（图 4-25）个元胞为邻居的类型。除此之外，也可以根据题目需要自行设计邻居的形式。

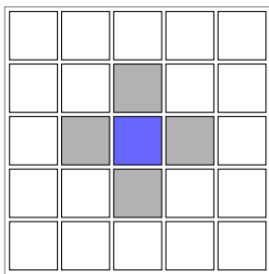


图 4-23

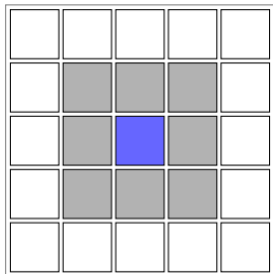


图 4-24

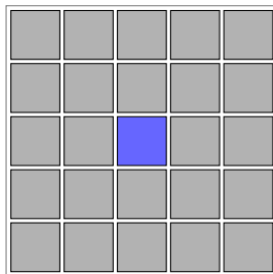


图 4-25

### 4. 边界条件

在理论上，元胞空间通常是在各维向上且无限延展的，这有利于在理论上的推理和研究。但是在实际应用过程中，我们无法在计算机上实现这一理想条件，因此，实际模拟中就存在边界。在定义元胞的邻居时，会出现一个问题，即边界上元胞的邻居不足。元胞自动机对每个元胞施加同样的规则，而边界上的元胞与非边界上的元胞具有不同的邻居数。因此需要设定不同的边界条件，使边界上的元胞也与正常的元胞具有相同的邻居数量。边界条件主要有四种类型：定值型，周期型，反射型和吸收型，如图 4-26 所示。其中较为常用的是周期型边界，周期型是指相对边界连接起来的元胞空间。对于一维空间，元胞空间表现为一个首尾相接的“圈”。对于二维空间，上下相接，左右相接。而形成一个拓扑圆环





面，形似车胎或甜点圈。周期型空间与无限空间最为接近，因而在理论探讨时，常以此类空间型作为试验。图 4-27 所示为二维方型网格周期边界条件时，三种元胞 ( $a$ ,  $b$  和  $c$ ) 的 Moore 邻居。元胞  $a$  为非边界元胞，它有 8 个正常的邻居，而  $b$  和  $c$  是边界元胞，它们没有足够的邻居，通过周期性边界条件，可以规定图 4-27 所示的邻居。

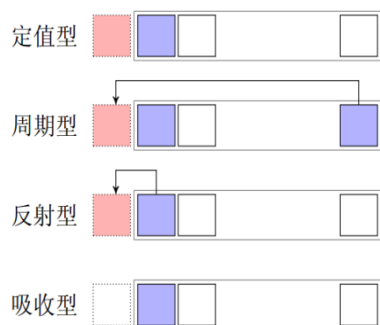


图 4-26

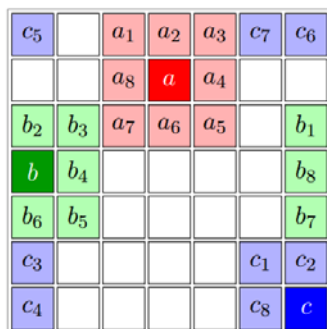


图 4-27

## 5. 规则

根据元胞当前状态及其邻居状况确定下一时刻该元胞状态的动力学函数，简单讲，就是一个状态转移函数。我们称该函数为元胞自动机的局部映射或局部规则。规则是支配整个元胞自动机的动力学行为的。一般规则都是定义在局部。通过局部间元胞的相互作用而引起全局变化。比如说流行病的传播，只有相互靠得近的人才能进行直接的传播，这就是局部作用，但当流行病传播一定时间时就会引起大范围的变化，这就是全局变化。而这个全局变化是由一个个局部作用引起的。元胞自动机的规则一般为总和型。就是说某个元胞下时刻的状态只决定于它所有邻居当前状态及自身的当前状态。一个简单的例子是足球场观众台上的人浪，当你最靠近的人即邻居站起来时，你才能站，当他们坐下后你才能坐下，这就是规则。

### 4.3.3 应用举例

本节将主要介绍两个元胞自动机模型及其 MatLab 程序实现，使读者进一步深入了解元胞自动机模型，并掌握简单的元胞自动机模型编程的实现方法。

#### 1. 森林火灾元胞自动机规则介绍

根据模拟的需要，元胞自动机规则中还可以引入概率，一般把规则中含有概率的元胞自动机称为概率元胞自动机（概率机）。一个最简单的二维概率元胞自动机是森林火灾模型，该模型的规则是由 Drossel 和 Schwabl<sup>[11]</sup>在 1992 年给出的，因此该模型又称 Drossel-Schwabl 森林火灾模型。森林火灾元胞自动机模型定义在正方网格上，元胞有三种状态：树（未燃烧的树），火（正在燃烧的树）和空状态。某元胞下时刻状态由该时刻本身的状态和周围四

个邻居（VonNeumann 邻居）的状态决定。具体规则：

（1）着火：若某元胞的状态为“树”，且其最邻近的 4 个邻居元胞中有状态为“火”的元胞，则该元胞下时刻的状态由“树”变为“火”，用以模拟火势的蔓延。对于邻居中没有“火”的情况，状态为“树”的元胞也会以一个低概率  $f$  变为“火”，用以模拟闪电引起的火灾。其着火规则如图 4-28 所示。

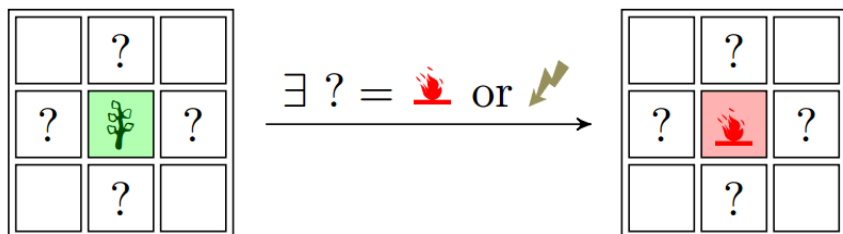


图 4-28

（2）烧尽：状态为“火”的元胞，其状态在下一时刻将变成“空”，用以模拟着火的树经过一定时间后被烧完。其烧尽规则如图 4-29 所示。



图 4-29

（3）新生：状态为“空”的元胞，在下一时刻以一个低概率  $p$  变为“树”，用以模拟新树的长出。其新生规则如图 4-30 所示。

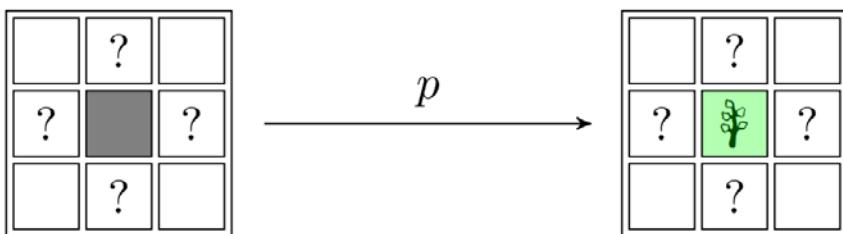


图 4-30

以上的三条规则中，着火规则中的闪电以及新生规则都涉及概率，并体现了该模型在演化过程中出现的随机性和不确定性。图 4-31 所示是森林火灾元胞自动机模拟效果图。

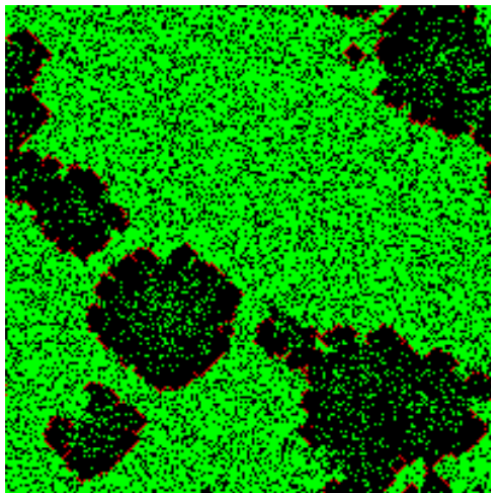


图 4-31

假设元胞空间足够大以至于没有临界作用产生，在周期性边界条件下，开始于一个任意的原始条件，系统经过一段时间后达到稳定状态，此时系统可由燃烧着树的密度  $\rho_f$ ，空地密度  $\rho_e$ ，树的密度  $\rho_t$  三个参数值描述。这三个参数之和等于 1，即

$$\rho_f + \rho_e + \rho_t = 1$$

系统达到稳定状态时，正在生长的树的数量与正在燃烧中的树的数量相等，即：

$$\rho_f = p\rho_e$$

树木大量生长的前提条件是燃烧着的树的密度  $\rho_f$  很小。若  $\rho_f$  很大时，树不可能成长为森林。式子表明只有当树生长的速率  $p$  接近于 0 时发生火灾的概率才会很小。为了保证大片森林的形成，闪电的概率  $f$  还必须满足  $f < p$ ，当闪电击中一小丛树，而且没有树在它们周围时它们会烧的很快。但是，如果闪电击中了一大丛树时，将需要一段甚至很长的时间去燃烧，而且在燃烧的同时周边地区还有一些树木生长，这样的火灾很难扩大。为了观察其中关键的自我复制的行为，大树丛与小树丛必须以同样的方式去燃烧。当取树木的生长速率  $p$  足够小时，即使是大规模的树林在周围没有新树生长的条件下也可以迅速的燃烧。在这种情况下，系统的变化取决于  $f/p$ ，而不是  $f$  和  $p$  独立值。 $f/p$  实际上给出了平均两次闪电间新生出的树的数量<sup>[12,13]</sup>。当  $f$  和  $p$  以同样的比例减小时，系统整体会以这种比例进行变化，而不依赖两次闪电发生的时间间隔中生长的树的数量，这种情形下森林迅速燃烧的方式可以用  $p < T_{s\max}$  表示，其中  $T_{s\max}$  表示大规模森林燃烧需要的时间。综上所述，概率元胞自动机在森林火灾模型中的运用条件为：

$$f < p < T_{s\max}$$

其中：



$f$ ——闪电引起树变为火的概率;  
 $P$ ——长出新树的概率;  
 $T_{s\max}$ ——大规模森林燃烧需要的时间。

## 2. 森林火灾元胞自动机程序实现

利用 MatLab 强大的逻辑运算功能, 森林火灾元胞自动机很容易实现, 森林火灾元胞自动机模拟效果图是由以下程序代码执行后得到的, 代码如下:

```
n = 200;           % 表示森林矩阵的尺寸: n x n
Pltg = 5e-6;       % 闪电的概率
Pgrw = 1e-2;       % 生长的概率
NW = [n 1:n-1];    % 用于构造北邻居 veg(NW,:) 和西邻居 veg(:,NW)
SE = [2:n 1];      % 用于构造南邻居 veg(SE,:) 和东邻居 veg(SE,:)
veg = zeros(n);     % veg = {0 表示空, 1 表示火, 2 表示树}

imh = image( cat(3,(veg==1),(veg==2),zeros(n)) );
for i=1:3000        % 周围四个邻居中状态为火的数量
    num =           (veg(NW,')==1) + ...
                    (veg(:,NW)==1) + (veg(:,SE)==1) + ...
                    (veg(SE,')==1);
    veg = 2*( (veg==2) | (veg==0 & rand(n)<Pgrw) ) - ...
          ( (veg==2) & (num >0 | rand(n)<Pltg) );
    set(imh, 'cdata', cat(3,(veg==1),(veg==2),zeros(n)) );
    drawnow
end
```

以上程序构造了一个  $n \times n$  的数组 `veg` 来表示一片森林, 数组中分别 0, 1, 2 表示空, 火, 树三种元胞状态。在周期边界条件下, 向量 `NW` 用于构造所有元胞的北邻居 `veg(NW, :)` 和西邻居 `veg(:, NW)`, `SE` 用于构造南邻居 `veg(SE, :)` 和东邻居 `veg(:, SE)`。程序的 11 至 13 行求得所有元胞东南西北四个邻居中为火的数量 `num`。比如 `num(i, j)` 就表示在 `veg` 中第  $i$  行  $j$  列的元胞周围四个元胞中状态为火的数量。程序的 15, 16 行是森林元胞自动机规则的体现:

- 第 15 行程序可以分解成  $2 \times (\text{veg} == 2)$  和  $2 \times (\text{veg} == 0 \& \text{rand}(n) < \text{Pltg})$  两部分, 这两部分分别表示了 `veg` 中某元胞下一时刻的状态为树的两种前提条件: 要么该元胞本身的状态就是树; 要么该元胞的状态为空, 但满足生长概率。
- 第 16 行程序可以分解为  $\text{veg} == 2 \& \text{num} > 0$  和  $\text{veg} == 2 \& \text{rand}(n) < \text{Pgrw}$  两部分, 这两部分分别表示了 `veg` 中某元胞下一时刻状态为火的两种条件: 当前时刻该元胞状态为树, 且周围四个邻居中有状态为火的元胞; 当前时刻该元胞状态为树, 且该元胞满足闪电的概率。

这两行程序很巧妙地用减号连接了起来, 当第 16 行程序的逻辑表达式为真时, 则第 15 行的逻辑表达式的结果会被减去 1, 恰好实现了元胞状态从树变为火的转变。需要注意

的是第 15 行、第 16 行的程序中隐含了一种情况：对于 veg 中为 1（火）的元胞下一时刻变为 0（空）。第 7 行程序是将矩阵可视化，用黑色，红色和绿色的格子分别表示状态为空，火和树的元胞。第 18 行程序则是更新图形的颜色数据。

### 3. 交通流 NS 元胞自动机规则介绍

在第 2 节中已经给出了 184 号交通流元胞自动机模型，184 号规则虽然简单，但也能用来模拟数学建模中的某些交通问题，在 2005 年 MCM 的比赛中，Duke 大学的学生基于 184 号规则，对收费站问题进行了模拟，并获得了当年的特等奖。不过大多情况下，用 184 号规则来模拟交通问题还是过于简单，很难捕捉复杂的交通现象。1992 年，德国物理学家 Kai Nagel 和 Michael Schreckenberg 在 184 号规则基础上提出了一维交通流 Nagel Schreckenberg(NS)模型（又称 NS 规则）<sup>[14]</sup>。NS 规则是最常用的交通流模型之一，该模型既简单又能模拟出交通堵塞的特征，比如能够展现出当交通拥挤时，车辆的平均速度变小的行为。单行道 NS 模型也是一个概率元胞自动机，每辆车的状态都由它的速度和位置所表示。图 4-33 所示是 NS 模型的四条规则的示意图，初始时状态四辆车的位置分别为  $x_1=1, x_2=3, x_3=6, x_4=7$ ，速度分别为  $v_1=2, v_2=1, v_3=1, v_4=0$ 。NS 模型的四条规则分别为：

- 加速规则：司机总是期望以最大的速度行驶，如果第  $n$  辆车的速度未达到最大速度，则其速度增加 1： $v_n \rightarrow \min(v_n + 1, v_{\max})$ ，系统允许的最大速度为  $v$ ，经过加速后，图 4-32 中四辆车的速度分别变为  $v_1=2, v_2=2, v_3=2, v_4=1$ 。
- 防止碰撞：为避免与前车碰撞，若第  $n$  辆车的速度  $v_n$  超过了前方空元胞数  $d_n$ ，则其速度强制降为  $d_n - 1$ ： $v_n \rightarrow \min(v_n, d_n - 1)$ 。经过防止碰撞规则后，图 4-32 中四辆车的速度分别变为  $v_1=1, v_2=2, v_3=0, v_4=1$ 。

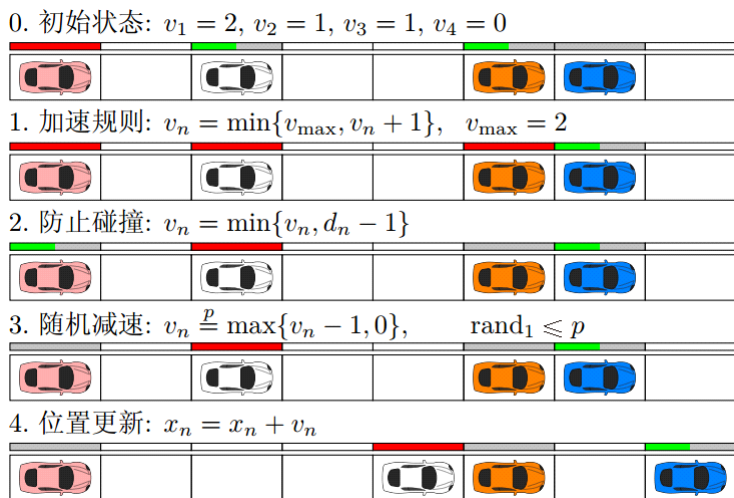


图 4-32



- 随机减速：由于司机的过度反应以及诸多不确定因素以，都会造成的车辆的减速。若第  $n$  辆车的速度  $v_n > 1$ ，则  $v_n$  以一个较低的概率降低 1： $v_n \rightarrow \max(v_n - 1, 0)$ 。经过随机减速后，图 18 中四辆车的速度分别变为  $v_1 = 0$ ， $v_2 = 2$ ， $v_3 = 0$ ， $v_4 = 1$ 。
- 位置更新：模拟车的前进，若第  $n$  辆车的速度为  $v_n$ ，位置为  $x_n$ ，则下一时刻的位置为  $x_n = x_n + v_n$ ，经过位置更新后，图 4-32 中四辆车的速度分别变为  $v_1 = 1$ ， $v_2 = 5$ ， $v_3 = 6$ ， $v_4 = 8$ 。

用计算机无法直接模拟一条无限长的车道，通常用周期性边界构造一条环道来模拟车流密度固定的无限长车道的交通情况。也有应用开放边界条件，在车道的一端不断地有随机生成的车辆流入，当车辆通过车道的另一端则从系统中移除。如果没有特殊考虑，周期性边界就能很好的满足模拟的要求。相对开放边界条件，周期性边界条件（封闭环行车道）更为简单，更容易控制和固定车流密度，平均速度和流量也不会有太大波动，便于测试不同密度下的各种参量。日本及 MIT 的学者对封闭环形车道内的交通情况都做过实验<sup>[15]</sup>和研究，非常有意思，有相关的报道<sup>[16, 17]</sup>和实验视频<sup>[18]</sup>及动画<sup>[19]</sup>。因此有一些现象和结论可以供参考和对比。图 4-33 所示为在周期性边界条件（环形车道）下，交通流的元胞自动机 NS 模型的模拟效果图（为了出图效果，效果图中的道路长度被降为 36）。图 4-33 中形成阻塞的现象与实验视频<sup>[18]</sup>及动画<sup>[19]</sup>非常相似。

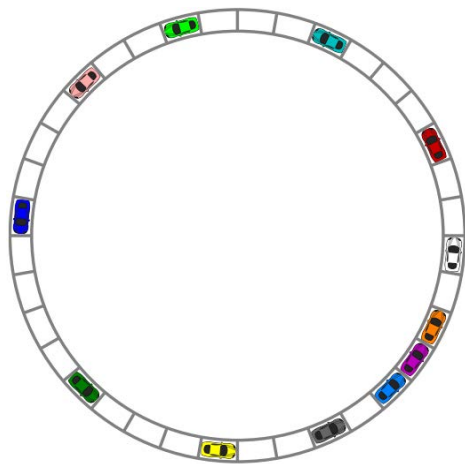


图 4-33

如果将随机减速的概率设置为 0，去掉 NS 规则中的随机因素，则通过简单的分析就可以得到单行道元胞自动机的 NS 模型的一些性质。比如系统平衡时，系统的平均速度和平均流量<sup>[20]</sup>。

$$v = \min(v_{\max}, d), J = \min(\rho v_{\max}, 1 - \rho)$$

具体分析过程可见文献<sup>[20]</sup>。通过模拟，也可以发现随机减速的概率  $p_{\text{brak}}$  设为 0 时，系统最终会达到一种平均，即系统以一定的周期做循环运动，系统的平均速度也不再改变。



图 4-34 所示是密度与流量的观测值及模拟值的对比, 图 4-34 (a) 所示为观测值<sup>[21]</sup>, 图 4-34 (b) 中的点为模拟得到的值, 线上数据是由上式给出的理论值。

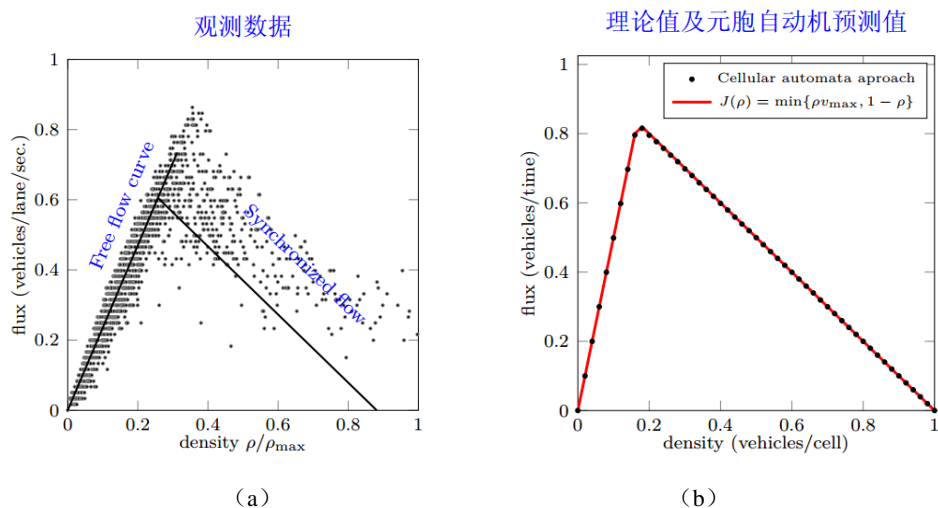


图 4-34

图 4-34(b)中模拟结果是由一条长为  $L=100$  的封闭环形车道、车辆的最大速度  $v_{\max}=5$ 、改变车流密度  $\rho \in [0,1]$  并经每种密度情况下模拟多次且测试不同密度情况下的平均流量和平均速度得到的。此外, 这里还对比了航拍得到的时空轨迹图<sup>[15]</sup>和模拟得到的时空曲线图(随机减速的概率  $p_{\text{brak}}=0.15$ ), 如图 4-35 所示。通过图 4-34 和图 4-35 所示, 可以进一步说明 NS 规则能合理的反映交通流的特点。

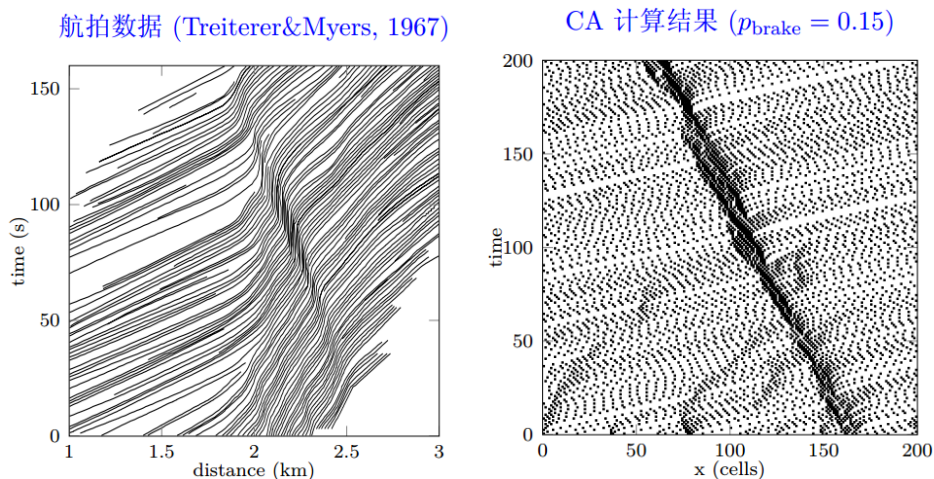


图 4-35

#### 4. 交通流 NS 元胞自动机程序实现

利用 MatLab 强大的数组运算, 单环形车道交通流的元胞自动机 NS 模型很容易实现,



可以用如下程序来实现 NS 模型的模拟并计算, 通过计算可求出平均流量等参数。

### NS 模型的模拟及计算

```
function [flux, vmean] = ns(rho, p, L, tmax)
% 输入: rho-密度; p-随机减速概率; L-环形车道长; tmax-模拟总时间步数
% 输出: flux-平均流量; vmean-平均速度
%
ncar = ceil(L*rho);           % 根据密度\rho 计算系统中车辆数
vmax = 5;                    % 车辆最大行驶速度
x = sort(randperm(L, ncar));  % 位置向量: 随机生所有车辆的位置, 并排序
v = vmax * ones(1, ncar);    % 速度向量: 初始化所有车辆的速度为
flux = 0; vmean = 0;         % 流量和平均速度: 初始化为零

h = plotcirc(L, x, 0.1);     % 绘制环形车道和车辆, 并返回车辆句柄
for t = 1:tmax
    v = min(v+1, vmax);      % 加速规则
    gaps = gaplength(x, L);
    v = min(v, gaps-1);      % 防止碰撞
    vdrops = ( rand(1,ncar)<p );
    v = max(v - vdrops, 0);   % 随机减速
    x = x + v;                % 位置更新
    passed = x>L;             % 找出超出边界的车辆
    x(passed) = x(passed) - L; % 周期边界
    if t>tmax/2
        flux = flux + sum(v)/L; % 计算流量
        vmean = vmean + mean(v); % 计算平均速度
    end
    plotcirc(L, x, 0.1, h);   % 更新图像, 以形成动画
end
flux = flux/(tmax/2);         % 流量对时间平均
vmean = vmean/(tmax/2);      % 平均速度对时间平均
```

以上 MatLab 程序定义了一个 ns 函数, 只需给出环形车道上的车辆密度, 随机减速概率, 环形车道长度和模拟时间步, 该函数就能计算出环道上的平均流量和流速。以上程序比较简单。仅以下几行需要额外解释一下:

(1) 第 7 行中的 randperm(L,ncar) 函数是从  $[1, 2, \dots, L]$  中随机抽取 ncar 个数, 以表示 ncar 辆车初始随机分布于环形车道上, 再由 sort 函数排序, 向量 x 中的车辆按位置从小到大排列。

(2) 第 24~25 行是实施周期性边界条件, 在经过第 22 行的位置更新后, 可能会有车辆的位置超出了 L, 对于位置超出 L 的车辆 (passed 为真的车辆) 需要减掉一个周期 L, 使车辆的位置仍然处于 1 到 L 之间。

(3) 第 27~30 行用来计算平均流量和平均速度, 前  $t_{\max}/2$  的时间用来让系统达到平衡, 平均流量和平均速度的计算仅在  $t > t_{\max}/2$  的时间里进行。这里是通过计算  $\rho \bar{v}$  来计算流量的。流量的计算也可通过选取一点, 计算单位时间通过的车辆数来计算, 比如计算





passed 为真的平均数量。

以上程序中还调用了两个自定义的函数 `gaplength` 和 `plotcirc`。函数 `gaplength` 用来计算相邻两边间的距离，其主要代码如下：

用 MatLab 实现函数 `gaplength` 代码

```
function gaps = gaplength(x, L)
% 输入: x-各车辆位置; L-环形车道长      输出: gaps-相邻两车间的距离
%
ncar = length(x);                % 获得系统中的车辆数
gaps = inf*ones(1, ncar);        % 初始化相邻两车间的距离
if ncar>1
    gaps = x([2:end 1]) - x;
    gaps(gaps<0) = gaps(gaps<0) + L;    % 周期性条件
```

上述程序中的第 8 行用于计算两邻车的距离。对于第 1 辆车， $\text{gaps}(1) = x(2) - x(1)$ ；对于第  $i$  辆车，则有  $\text{gaps}(i) = x(i+1) - x(i)$ ；周期性边界构成首尾相接的环形车道，对于最后一辆车  $\text{gaps}(\text{end}) = x(1) - x(\text{end})$ ，这时可能出现 `gaps` 中出现负值，第 9 行通过对 `gaps` 中小于 0 的值加一个周期  $L$  来保证所有车辆都有正确的与前车的距离。

函数 `plotcirc` 用来将数据图形化，动态地显示每辆车的位置，其主要代码如下：

用 MatLab 实现函数 `plotcirc` 代码

```
function h = plotcirc(L, x, dt, h)
% 输入: L-环形车道长; x-各车辆位置; dt-动画相邻两帧间隔时间; h-车辆句柄
% 输出: h-车辆句柄
%
W = 0.05; R = 1;                % 车道的宽度; 环形车道的半径
ncar = length(x);                % 获得系统中的车辆数

theta = 0:2*pi/L:2*pi;          % 环形车道上径向网格线的极角
xc = cos(theta); yc = sin(theta); % 径向网格线的中心
xin = (R-W/2)*xc; yin = (R-W/2)*yc; % 径向网格线内端点
xot = (R+W/2)*xc; yot = (R+W/2)*yc; % 径向网格线外端点

xi = [xin(x); xin(x+1); xot(x+1); xot(x)]; % 车辆所在网格的四个端点
yi = [yin(x); yin(x+1); yot(x+1); yot(x)]; %

if nargin == 3                    % 如果函数输入量等于 3, 用于首次绘图
    color = randperm(ncar);       % 为各车辆随机生成一个独立的颜色
    h = fill(xi,yi, color); hold on % 画出所有车辆
    plot([xin; xot], [yin; yot], 'k', 'linewidth', 1.5); % 画径向网格线
    plot([xin; xot], [yin; yot], 'k', 'linewidth', 1.5); % 画环形网格线
    axis image;
else                               % 如果函数输入量等于 4, 用于图像更新
    for i= 1:ncar                 % 更新图像中车辆的位置
        set(h(i), 'xdata', xi(:,i), 'ydata', yi(:,i));
```





```
end
end
pause(dt)           % 暂停 dt 秒
```

在以上程序中,第 17 行中的 `nargin` 为调用该函数时实际输入参数的个数。输入参数的个数可以为 3,也可以为 4,输入参数个数为 3 时,用于首次调用函数 `plotcirc` 来绘制车道和车辆的初始位置,并返回表示所有车辆的矩形的句柄  $h$ 。输入参数个数为 4 时,额外的一个参数为表示每个车辆的四边形的句柄  $h$ ,这种情况下无需重新绘制车道和车辆,只需通过第 25 行的 `set` 来更新车辆的位置。

## 4.4 启发式算法简介及其在数学建模中的应用

### 4.4.1 引言

启发式算法 (Heuristic Algorithm) 兴起于 20 世纪 80 年代,启发式算法是一种基于直观或经验的局部优化算法,人们常常把从大自然的运行规律或者面向具体问题的经验和规则中启发出来的方法称之为启发式算法。现在的启发式算法也不是全部来自自然的规律,也有来自人类积累的工作经验。这些算法包括禁忌搜索 (tabu search)<sup>[1]</sup>,模拟退火 (simulated annealing)<sup>[2]</sup>,遗传算法 (genetic algorithms)<sup>[3]</sup>,人工神经网络 (artificial neural networks)<sup>[4]</sup>,蚁群算法 (Ant Algorithm)<sup>[5]</sup>等,下面给出其中几种启发式算法的简单定义:

- 模拟退火:通过模拟物理退火过程搜索最优解的方法。
- 遗传算法:通过模拟自然进化过程搜索最优解的方法。
- 神经网络:模仿动物神经网络行为特征,进行分布式并行信息处理的算法数学模型。
- 蚁群算法:模仿蚂蚁在寻找食物过程中发现路径的行为来寻找优化路径的几率型算法。

启发式算法主要用于解决大量的实际应用问题。目前,这些算法在理论和实际应用方面得到了较大的发展。无论这些算法是怎样产生的,它们有一个共同的目标——求 NP 难题组合优化问题的全局最优解。虽然有这些目标,但 NP 难题理论限制它们只能以启发式的算法去求解实际问题。而启发式算法使得能在可接受的计算费用内去寻找尽可能好的解,但不一定能保证所得解的可行性和最优性,甚至在多数情况下,无法描述所得解与最优解的近似程度。启发式算法包含的算法很多,例如解决复杂优化问题的蚁群算法 (Ant Colony Algorithms)。有些启发式算法是根据实际问题而产生的,如解空间分解,解空间的限制等;另一类算法是集成算法,这些算法是诸多启发式算法的合成。现代优化算法解决组合优化问题,如 TSP (Traveling Salesman Problem) 问题, QAP (Quadratic Assignment Problem) 问题, JSP (Job-shop Scheduling Problem) 问题等效果很好<sup>[6]</sup>。

启发式算法是数学建模竞赛中最常用的算法之一。在近十年来的美国大学生数学建模竞赛 (MCM) 和中国大学生数学建模竞赛 (CUMCM) 中,几乎每两年就会出现适合或需



要应用启发式算法的赛题。表 4-15 所列为近十来年美国大学生数学建模竞赛中用到启发式算法的特等奖论文不完全统计。

表 4-15

年份题号	题 目	特等奖论文数
2003MCM-B	Gamma 刀治疗方案	2
2006MCM-A	灌溉洒水器的安置和移动	2
2006MCM-B	通过机场的轮椅	1
2007MCM-B	飞机座位方案	1
2008MCM-B	建立数独拼图游戏	1
2009MCM-A	交通环岛的设计	1
2011MCM-A	滑雪赛道优化设计	1
2014MCM-B	最佳大学体育教练	1

本节我们只介绍模拟退火算法和遗传算法，并以 TSP 问题为例介绍两种算法具体的应用方法。在介绍模拟退火算法和遗传算法之前，先介绍一下旅行商问题。

旅行商问题，即 TSP 问题又译为旅行推销员问题、货郎担问题，是数学领域中著名难题之一。假设有一个旅行商人要拜访  $n$  个城市，他必须选择所要走的路径，路径的限制是每个城市只能拜访一次，而且最后要回到原来出发的城市。路径的选择目标是要求求得的路径路程为所有路径路程之中的最小值。旅行商问题也是数图论中最著名的问题之一，即“已给一个  $n$  个点的完全图，每条边都有一个长度，求总长度最短的经过每个顶点正好一次的封闭回路”。该问题可以被证明具有 NP 计算复杂性，迄今为止，这类问题中没有一个找到有效算法。如果用枚举的算法 1，并假设计算机枚举含 24 个城市的 TSP 问题需要 1 秒，则对于不同城市数量的 TSP 问题的求解时间见表 4-16。虽然枚举的算法能给全局最优解，但随着问题规模的增大，TSP 问题的枚举算法计算开销也急剧增加，甚至已经让人无法忍受。表 4-17 中给出了 34 个省会/直辖市的经纬度，图 4-36 所示是相应的中国地图及模拟退火算法给出的结果，若以中国 34 个省会作为旅行商问题，枚举算法几乎已经无法计算。下面将以中国 34 个省会作为旅行商问题，分别介绍模拟退火算法和遗传算法的 MatLab 程序实现。

表 4-16

城市数量	24	25	26	27	28	29	30	34
计算时间	1s	24s	10m	4.3h	4.9d	136.5d	10.8y	$\infty$

表 4-17

编号	1	2	3	4	5	6	7	8	9
经度	39.54	31.14	39.09	29.32	45.45	43.52	41.5	40.49	38.02
纬度	116.3	121.29	117.1	106.32	126.41	125.19	123.24	111.5	114.28
编号	10	11	12	13	14	15	16	17	18
经度	37.52	36.38	34.48	34.16	36.03	38.2	36.38	43.48	31.51
纬度	112.3	117	113.4	108.54	103.49	106.16	101.45	87.36	117.18
编号	19	20	21	22	23	24	25	26	27



(续表)

编号	1	2	3	4	5	6	7	8	9
经度	32.02	30.14	28.11	28.41	30.37	30.39	26.35	26.05	23.08
纬度	118.5	120.09	113	115.52	114.21	104.05	106.42	119.2	113.15
编号	28	29	30	31	32	33	34		
经度	20.02	22.48	25	29.39	22.18	22.14	25.03		
纬度	110.2	108.2	102.4	90.08	114.1	113.35	121.31		

TSP 问题枚举的算法复杂度

- 以第一个城市为始终点，计算任意一条路径 $[1, i_2, \dots, i_n, 1]$ 的长度的基本运算为两两城市间距离求和，基本操作次数为 $n$ 。路径的条数为 $(n-1)!$ ，求和运算的总次数为 $(n-1) \times n = n!$ 。
- 比较所有路径以得到最短路径，需要比较的次数为 $(n-1)$ 。

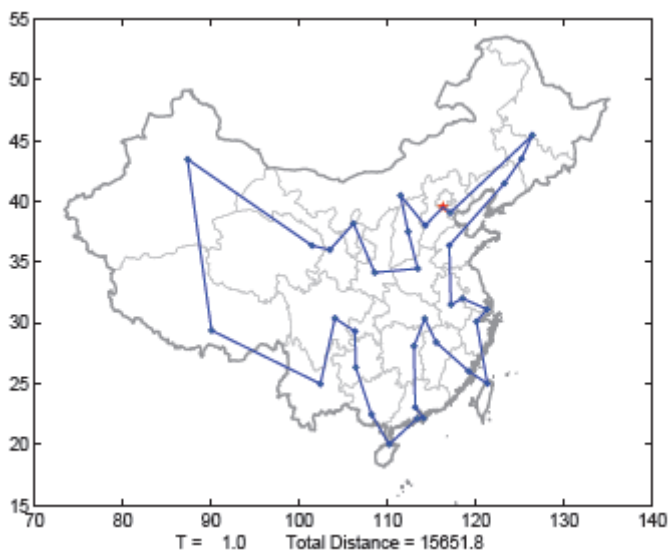


图 4-36

## 4.4.2 模拟退火算法

### 1. 算法介绍

模拟退火是一种通用概率算法，用来在固定时间内寻求在一个大的搜寻空间内找到的最优解。模拟退火是 S.Kirkpatrick, C.D.Gelatt 和 M.P.Vecchi 在 1983 年所发明。而 V.Cerny 在 1985 年也独立发明此算法<sup>2</sup>。模拟退火来自冶金学的专有名词退火。统计力学表明材料中粒子的不同结构对应于粒子的不同能量水平。在高温条件下，粒子的能量较高，可以自由运动和重新排列。在低温条件下，粒子能量较低。

退火是将金属材料加热后再经特定速率缓慢地冷却（这个过程被称为退火），粒子就



可以在每个温度下达到热平衡，目的是增大晶粒的体积，并且减少晶格中的缺陷。金属材料中的原子原来会停留在使内能有局部最小值的位置，加热使能量变大，原子会离开原来位置，而随机在其他位置中移动。退火冷却时速度较慢，使得原子有较多可能可以找到内能比原先更低的位置，最终形成处于低能状态的晶体。图 4-37 所示是物理退火示意图。

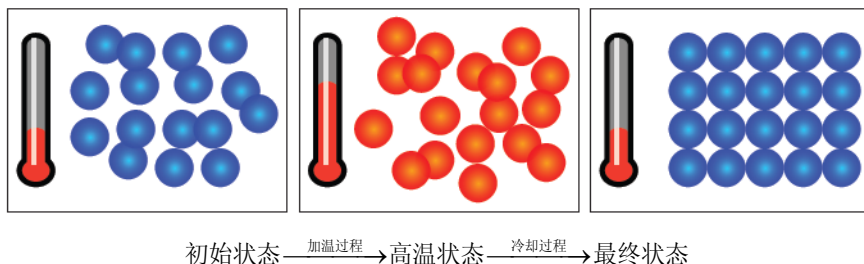


图 4-37

如果用粒子的能量定义金属材料的状态，Metropolis 法用一个简单的数学模型描述了退火过程。假设材料处于状态  $S_i$  时的能量为  $E(S_i)$ ，那么材料在温度  $T$  下从状态  $S_i$  转变为状态  $S_j$  概率为：

$$p(S_i \rightarrow S_j) = \begin{cases} 1 & E(S_i) \leq E(S_j) \\ \exp\left\{\frac{E(S_i) - E(S_j)}{K_B T}\right\} & E(S_i) > E(S_j) \end{cases}$$

其中  $K_B$  为波尔兹曼常数。式中表明如果  $E(S_i) \leq E(S_j)$ ，则接受状态的转变，否则以概率  $e^{[E(S_i) - E(S_j)] / (K_B T)}$  接受状态的转变。可以证明当温度降至很低时，材料会以很大概率进入最小能量状态<sup>[6]</sup>。

假定我们要解决的问题是一个寻找最小值的优化问题。考虑这样一个组合优化问题：优化函数为  $F: x \rightarrow R^+$ ，其中  $x \in S$ ，它表示优化问题的一个可行解， $R^+ = \{y | y \in R, y > 0\}$ ， $S$  表示函数的定义域。 $N(x) \subseteq S$  表示的一个邻域集合。将物理学中退火的思想应用于优化问题就可以得到模拟退火寻优方法。表 4-18 所列为模拟退火算法与物理退火过程的对应关系。首先给定一个初始温度  $T_0$  和该优化问题的一个初始解  $x_0$ ，并由生成下一个解  $x' \in N(x_0)$ ，是否接受  $x'$  作为一个新解依赖于下面概率

表 4-18

模拟退火	物理退火
解	粒子状态
目标函数	能量
最优解	能量最低态
设定初温	加温过程
扰动	热涨落
Metropolis 采样过程	热平衡,粒子状态满足波尔兹曼分布
控制参数的下降	冷却



$$p(x_0 \rightarrow x') = \begin{cases} 1 & f(x') \leq f(x_0) \\ \exp\left\{\frac{f(x') - f(x_0)}{T_0}\right\} & f(x') > f(x_0) \end{cases}$$

换句话说, 如果生成的解  $x'$  的函数值比前一个解的函数值更小, 则接受  $x'$  作为一个新解。否则以概率接受  $x'$  作为一个新解。泛泛地说, 对于某一个温度  $T_i$  和该优化问题的一个解  $x_k$ , 可以生成  $x'$ 。接受  $x'$  作为下一个新解的概率为

$$p(x_{k+1} \rightarrow x') = \begin{cases} 1 & f(x') \leq f(x_k) \\ \exp\left\{\frac{f(x') - f(x_k)}{T_0}\right\} & f(x') > f(x_k) \end{cases}$$

在温度  $T_i$  下, 经过很多次的转移之后, 降低温度  $T_i$ , 得到  $T_{i+1} < T_i$ 。在  $T_{i+1}$  下重复上述过程。因此整个优化过程就是不断寻找新解和缓慢降温的交替过程。最终的解是对该问题寻优的结果。我们注意到, 在每个  $T_i$  下, 所得到的一个新状态  $x_{k+1}$  完全依赖于前一个状态  $x_k$ , 可以和前面的状态无关, 因此这是一个马尔可夫过程。使用马尔可夫过程对上述模拟退火的步骤进行分析, 结果表明: 从任何一个状态  $x_k$  生成  $x'$  的概率, 在  $N(x_k)$  中是均匀分布的, 那么经过有限次的转换, 在温度  $T_i$  下的平衡态  $x_k$  的分布由下式给出:

$$p_k(T_i) = \frac{e^{-f(x_i)/T}}{\sum_{j \in S} e^{-f(x_j)/T_i}}$$

当温度降为 0 时,  $x_k$  的分布为:

$$p_k(0) = \begin{cases} 1/|S_{\min}| & x_k \in S_{\min} \\ 0 & \text{其他} \end{cases}$$

并且  $\sum_{x_k \in S_{\min}} p_k(0) = 1$ 。这说明如果温度下降十分缓慢, 而在每个温度都有足够多次的状态转移, 使之在每一个温度下达到热平衡, 则全局最优解将以概率 1 被找到。因此模拟退火算法所得解依概率收敛到全局最优解。

模拟退火算法新解的产生和接受可分为以下步骤:

- 由一个产生函数从当前解产生一个位于解空间的新解; 为便于后续的计算和接受, 减少算法耗时, 通常选择由当前新解经过简单地变换即可产生新解的方法, 如对构成新解的全部或部分元素进行置换, 互换等。注意到产生新解的变换方法决定了当前新解的邻域结构, 因而对冷却进度表的选取有一定的影响。
- 计算与新解所对应的目标函数差。因为目标函数差仅由变换部分产生, 所以目标函数差的计算最好按增量计算。事实表明, 对大多数应用而言, 这是计算目标函数差的最快方法。
- 判断新解是否被接受, 判断的依据是一个接受准则, 最常用的接受准则是



**Metropolis 准则:** 若  $\Delta t < 0$  则接受  $S$  作为新的当前解  $S$ , 否则以概率  $\exp(-\Delta t / T)$  接受  $S$  作为新的当前解  $S$ 。

- 当新解被确定接受时, 用新解代替当前解, 这只需将当前解中对应于产生新解时的变换部分予以实现, 同时修正目标函数值即可。此时, 当前解实现了一次迭代。可在此基础上开始下一轮试验。而当新解被判定为舍弃时, 则在原当前解的基础上继续下一轮试验。

模拟退火算法与初始值无关, 算法求得的解与初始解状态  $S$  (是算法迭代的起点) 无关; 模拟退火算法具有渐近收敛性, 已在理论上被证明是一种以概率 1 收敛于全局最优解的全局优化算法; 模拟退火算法具有并行性。

对于特定的问题, 在应用和设计模拟退火算法过程中应注意以下问题。

- **初始解的生成:** 通常是以一个随机解作为初始解。并保证理论上能够生成解空间中任意的解。也可以是一个经挑选过的较好的解, 这种情况下, 初始温度应当设置的较低。初始解不宜“太好”, 否则很难从这个解的邻域跳出。
- **邻解生成函数:** 邻解生成函数应尽可能保证产生的候选解能够遍布解空间。邻域应尽可能的小, 能够在少量循环步中充分探测, 但每次的改变不应该引起太大的变化。
- **确定初始温度:** 初始温度应该设置的尽可能的高, 以确保最终解不受初始解影响。但过高又会增加计算时间。在正式开始退火算法前, 可进行一个升温过程确定初始温度: 逐渐增加温度, 直到所有的尝试运动都被接受, 将此时的温度设置为初始温度。
- **确定等温步数:** 等温步数也称 **Metropolis 抽样稳定准则**, 用于决定在各温度下产生候选解的数目。通常取决于解空间和邻域的大小, 等温过程是为了让系统达到平衡, 因此可通过检验目标函数的均值是否稳定 (或连续若干步的目标值变化较小) 来确定等温步数。等温步数受温度的影响。高温时, 等温步数可以较小, 温度较小时, 等温步数要大。随着温度的降低, 增加等温步数。有时为了考虑方便, 也可直接按一定的步数抽样。
- **确定降温方式:** 理论上, 降温过程要足够缓慢, 要使得在每一温度下达到热平衡。但在计算机实现中, 如果降温速度过缓, 所得到的解的性能会较为令人满意, 但是算法会太慢, 相对于简单的搜索算法不具有明显优势。如果降温速度过快, 很可能最终得不到全局最优解。因此使用时要综合考虑解的性能和算法速度, 在两者之间采取一种折衷。

## 2. 程序实现

在前面已经简单的介绍了模拟退火算法主要思想。下面以 TSP 问题为例, 详细分析其 MatLab 程序实现。求解 TSP 问题的模拟退火算法描述如下。

- **解空间:** 如果我们按照表 4-17 中用整数 1~34 对每个城市进行编号, 解空间  $S$  表





示为  $\{1, 2, \dots, 34\}$  的所有固定起点和终点的循环排列集合。任何一种  $\{1, 2, \dots, 34\}$  的循环排列

$$X = \{(x_1, x_2, \dots, x_{34}) \mid x_i \in [1, 2, \dots, 34], x_i \neq x_j\}$$

都是本问题的一个解。其中  $x_i$  表示第  $i$  次访问的城市编号。

- 目标函数：TSP 问题的目标函数（或称代价函数）为路径长度，TSP 问题的目标为最小化路径长度，即

$$\min d(x_1, x_{34}) + \sum_{i=1}^{33} d(x_i, x_{i+1})$$

其中  $d(x_i, x_{i+1})$  表示编号为  $x_i$  和编号为  $x_{i+1}$  的两城市间的距离。

- 生成邻解：从  $\{1, 2, \dots, 34\}$  中随机抽取两个数  $i, j$ ，且有  $i < j$ ，将两个城市间的子路径逆向排序顺序生成新解

$$X' \leftarrow (x_1, x_2, \dots, x_j, x_{j-1}, \dots, x_{i+1}, x_i, \dots, x_{34})$$

在程序中还给出了生成邻解另一种方式‘对调’。

- 降温方式：利用选定的降温系数  $\alpha$  进行降温，即每执行 100 次循环降温一次：  
 $T \leftarrow \alpha T$  得到新的温度。
- 结束条件：用选定的终止温度 1.0，若  $T < 1.0$  计算终止，输出当前状态。

基于以上描述，可用 MatLab 实现 TSP 问题的模拟退火算法，模拟退火算法结果如图 4-40 所示。对于 34 个城市，可以定义一个结构数组 city，结构数组长度为 34，对于任何一个城市  $i$ ，可以用 city(i).lat 和 city(i).long 分别访问其经纬度。比如对于北京（见表 4-17 中标号为 1 的城市）有 city(1).lat=39.54, city(1).long=116.28。下面给出主程序的主要代码：

```
1 numberofcities = length(city); % 程序的数量
2 dis = distancematrix(city); % 距离矩阵 dij = dis(i, j)
3 temperature = 1000; % 初始温度
4 cooling_rate = 0.94; % 降温速度
5
6 route = randperm(numberofcities); % 初始化路径
7 previous_distance = totaldistance(route, dis); % 计算路径的长度
8 temperature_iterations = 1; % 用于计算恒温步数
9
10 while 1.0 < temperature
11     temp_route = perturb(route, 'reverse'); % 生成新解(邻解)
12     current_distance = totaldistance(temp_route, dis); % 计算新解的距离
13     diff = current_distance - previous_distance; % 代价函数差
14
15     % Metropolis 准则
16     if (diff < 0) || (rand < exp(-diff/(temperature))) % 接受新解的条件
```



```

17 route = temp_route;
18 previous_distance = current_distance;
19
20 temperature_iterations = temperature_iterations + 1;
21 end
22
23 % 每 100 步降一次温
24 if temperature_iterations _ 100
25     temperature = cooling_rate*temperature; % 降温方式:  $T = \_ \_ T$ 
26     temperature_iterations = 0;
27 end
28
29 end

```

以上程序中调用了函数 `distancematrix`, `totaldistance`, `perturb`, 下面我们逐一介绍。

函数 `distancematrix` 用来求得任意两城市间的球面距离矩阵 `dis`。由于城市的坐标由经度、纬度表示, 两城市间的距离需要用球面距离来表示, 而 MatLab 中自带的函数 `distance` 可用来求取球面上两点的球面距离。函数 `distancematrix` 的代码如下:

```

1 function dis = distancematrix(city)
2 numberofcities = length(city);
3 R = 6378.137; % 地球半径
4 for i = 1: numberofcities
5     for j = i+1: numberofcities
6         dis(i,j) = distance(city(i).lat , city(i).long , ...
7                             city(j).lat , city(j).long , R);
8         dis(j,i) = dis(i,j);
9     end
10 end

```

函数 `distancematrix` 返回距离矩阵 `dis`, 其中 `dis(i,j)` 表示第  $i$  个城市到第  $j$  个城市的距离。

函数 `totaldistance` 用来计算一条路径的总长度。如果用一个向量 `route` 来表示一条路径 (问题的解), 向量 `route` 是  $\{1, 2, \dots, 34\}$  的某种循环排列。可以定义以下函数 `totaldistance` 来非常方便和快速的求取一条路径的总长度。

```

1 function d = totaldistance(route , dis)
2 d = dis(route ( e n d ) , route (1)); % 形成圈
3 for k = 1: length(route) -1
4     i = route(k);
5     j = route(k+1);
6     d = d + dis(i,j); %  $d = \text{dis}(1; 34) + \sum_{i=1}^{34} \text{dis}(i, i+1)$ 
7 end

```





其中距离矩阵  $\text{dis}$  是由函数 `distancematrix` 求得的。函数 `totaldistance` 返回路径  $\text{route}$  的总长度  $d$ ，TSP 问题的目标就是  $\min d$ 。

函数 `perturb` 用来从当前解产生一个位于解空间的新解。函数 `perturb` 的代码如下：

```
1 function route = perturb(route , method)
2 numbercities = length(route); % 城市数量
3 i = randsample(numbercities ,1);% 随机整数满足  $i \in [1, 2, \dots, \text{numbercities}]$ 
4 j = randsample(numbercities ,1);% 随机整数满足  $j \in [1, 2, \dots, \text{numbercities}]$ 
5 switch method
6     case 'reverse' % 逆序
7         citymin = min(i,j);
8         citymax = max(i,j);
9         route(citymin:citymax) = route(citymax :-1: citymin);
10    case 'swap' % 对调
11        route ([i, j]) = route ([j, i]);
12 end
```

函数 `perturb` 定义了两种生成邻解的方式，一种是‘逆序’，另一种为‘对调’。当参数 `method` 为‘reverse’时，生成邻解的方式为‘逆序’，随机选择路径中的两个城市，并将两个城市间的子路径逆向排序顺序生成新解；当参数 `method` 为‘swap’时，生成邻解的方式为‘对调’。随机选择路径中的两个城市，并将两个城市间的子路径逆向排序顺序生成新解。

### 4.4.3 遗传算法

#### 1. 算法介绍

遗传算法（Genetic Algorithm）是由美国的 J·Holland 教授于 1975 年首先提出的，是计算数学中用于解决最优化的搜索算法。遗传算法最初是借鉴了进化生物学中的一些现象而发展起来的，这些现象包括遗传，突变，自然选择（适者生存，优胜劣汰遗传机制）以及杂交等。在遗传算法中，问题域中的可能解被看作是群体的个体。对于一个最优化问题，一定数量的候选解（称为个体）的抽象表示（称为染色体）的种群向更好的解进化。传统上，解用二进制将个体编码成符号串形式（即 0 和 1 的串），但也可以用其他表示方法。进化从完全随机个体的种群开始，之后一代一代发生。在每一代中，整个种群的适应度被评价，基于它们的适应度，从当前种群中随机地择优选择多个个体。通过杂交和突变产生新的生命种群，该种群在算法的下一代迭代中成为当前种群。从而不断得到更优的群体，同时搜索优化群体中的最优个体，求得满足要求的最优解。

遗传算法可分为以下基本步骤。

（1）初始化：初始化进化代数计数器  $t \leftarrow 0$ ，最大进化代数  $T$ 。随机生成  $M$  个个体作为初始体  $P(t)$ ，始群体  $P$ 。



- (2) 个体评价：计算  $P(t)$  中各个个体的适应度值。
- (3) 选择运算：将选择算子作用于群体。
- (4) 交叉运算：将交叉算子作用于群体。
- (5) 变异运算：将变异算子作用于群体，并通过以上运算得到下一代群体  $P(t+1)$ 。
- (6) 终止条件：如果  $t \leq T$ ，则  $t \leftarrow t+1$  并跳转到第 2 步；否则输出  $P(t)$  中的最优解。

基本遗传算法的五个组成部分如下。

- 编码：正如研究生物遗传是从染色体着手，而染色体则是由基因排成的串，遗传算法中，首要问题就是如何通过某种编码机制把对象抽象为由特定符号按一定顺序排成的串（解的形式）。编码影响到交叉，变异等运算，很大程度上决定了遗传进化的效率。在基本遗传算法（SGA）使用上，二进制串进行编码，每个基因值为符号 0 和 1 所组成的二进制数。针对不同的问题，可以适当选择其他编码形式，如格雷编码，实数编码，符号编码。
- 适应度函数：适应度函数也称评价函数。是根据目标函数确定的用于区分群体中个体好坏的标准。适应度函数是遗传算法进化过程的驱动力，也是对个体的优胜劣汰的唯一依据。它的设计应结合求解问题本身的要求而定。一般情况下适应度是非负的，并且总是希望适应度越大越好（适应度值与解的优劣成反比例）。通常适应度函数可以由目标函数直接或间接改造得到。比如，目标函数或目标函数的倒数、相反数经常被直接用作适应度函数。适应度函数不应过于复杂，以便于计算机的快速计算。
- 选择算子：选择运算的使用是对个体进行优胜劣汰；从父代群体中选取一些适应度高个体，遗传到下一代群体。适应度高的个体被遗传到下一代群体中的概率大；适应度低的个体，被遗传到下一代群体中的概率小。基本遗传算法中选择算子采用轮盘赌选择方法，图 4-38 所示为轮盘赌示意图。轮盘赌又称比例选择算子，个体  $i$  被选中的概率  $p_i$  与其适应度成正比： $p_i = f_i / \sum_{j=1}^N f_j$ 。个体的适应度值越大，

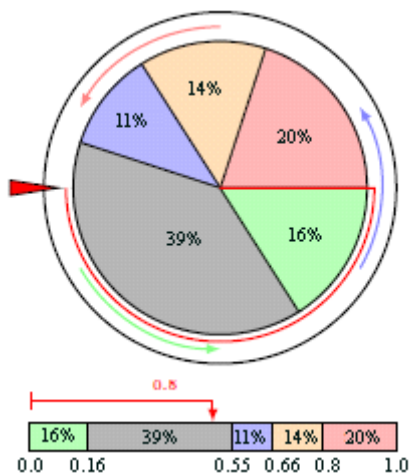


图 4-38



被选中的概率就越高，直接体现了“适者生存”这一自然选择原理。遗传算法中也常用其他选择算子，如两两竞争（从父代中随机地选取两个个体，比较适应值，保存优秀个体，淘汰较差的个体）等。

- 交叉算子：交叉运算是指对两个相互配对的染色体依据交叉概率按某种方式相互交换其部分基因，从而形成两个新的个体。交叉运算是遗传算法区别于其他进化算法的重要特征，它在遗传算法中起关键作用，是产生新个体的主要方法。遗传算法中，交叉算子可以是单点交叉，也可以是多点交叉，其示意图如图 4-39 所示。基本遗传算法中交叉算子采用单点交叉算子。

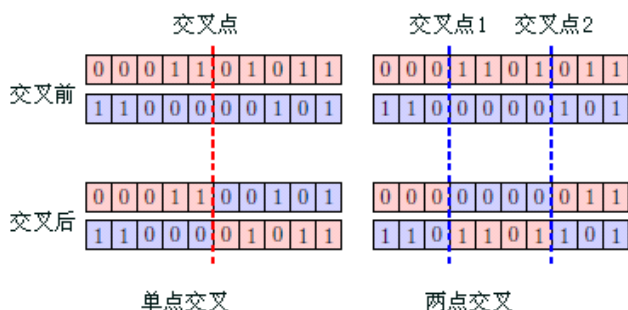


图 4-39

- 变异算子：变异操作对群体中的个体的某些基因座上的基因值作变动，模拟生物在繁殖过程，新产生的染色体中的基因会以一定的概率出错。变异运算是产生新个体的辅助方法，决定遗传算法的局部搜索能力，保持种群多样性。交叉运算和变异运算的相互配合，共同完成对搜索空间的全局搜索和局部搜索。遗传算法中，变异算子可以是基本位变异，也可以是换位变异，如图 4-40 所示。基本遗传算法中变异算子采用基本位变异算子。基本位变异算子是指对个体编码串随机指定的某一位或某几位基因作变异运算。对于二进制编码符号串所表示的个体，若需要进行变异操作的某一基因座上的原有基因值为 0，则将其变为 1；反之，若原有基因值为 1，则将其变为 0。

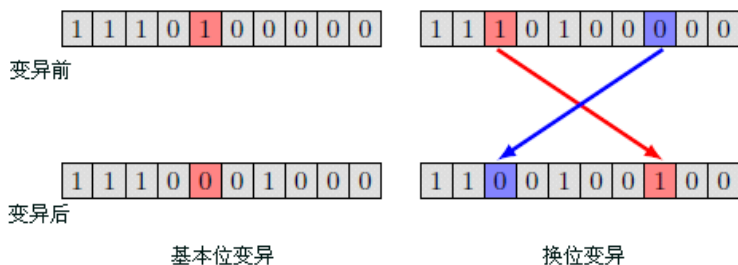


图 4-40

## 2. 程序实现

在前面已经简单的介绍了模拟退火算法主要思想。在下面以 TSP 问题为例，详细分析其 MatLab 程序实现。求解 TSP 问题的遗传算法描述如下。

- 编码：如果我们按照表 4-17 用整数 1~34 对每个城市进行编号，则任何一种  $\{1, 2, \dots, 34\}$  的循环排列都是本问题的一个解。对于 TSP 问题，选用符号编码，直接用  $\{1, 2, \dots, 34\}$  的循环排列作为解的符号串形式。
- 适应度函数：TSP 问题的目标函数为路径长度  $d$ ，考虑到适应度函数的特点，可将路径长度的倒数  $1/d$  作为适应度函数。
- 选择算子：轮盘赌。
- 交叉算子：采用两点交叉，采用符号编码的解形式在交叉后的必须要保证生成的新解也是一个合法的  $\{1, 2, \dots, 34\}$  的循环排列，不能出现重复的城市编号。因此在随机选取两个交叉点  $i$  和  $j$  后 ( $i < j$ ) 并进行交叉（交换双亲第  $i$  个到第  $j$  个符号串）后，还需要一个额外的操作以消除重复的城市编号。如果交换后，后代 1 到  $i$  中的编号  $x_k$ ， $1 < k < i$  与子串  $i$  到  $j$  中的编号  $x_m$ ， $i \leq m \leq j$  重复，则用交叉前的  $x_m$  来替换到  $x_k$ ，按照此法对  $i$  到  $j$  中的编号依次作这样的操作，直到新解中不出现编号的重复为止。图 4-41 所示是这种交叉运算的结果示意图。

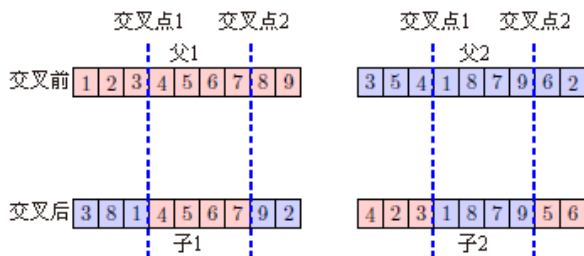


图 4-41

- 变异算子：在 TSP 问题中，单点基本位变异会造成新解中出现编号的重复。因此这里的变异算子采用对调，滑移和逆序三种不会引起编号重复的变异算子。

基于以上描述，可用 MatLab 实现 TSP 问题的遗传算法，图 4-42 所示为 MatLab 实现的中国 34 个省会城市 TSP 问题的遗传算法求解结果。在遗传算法的 MatLab 程序中，解的形式，任意两城市间的球面距离矩阵和路径的总长度的计算都与退火方法一致，相同的内容这里不再复述。下面给出主程序的主要代码如下。

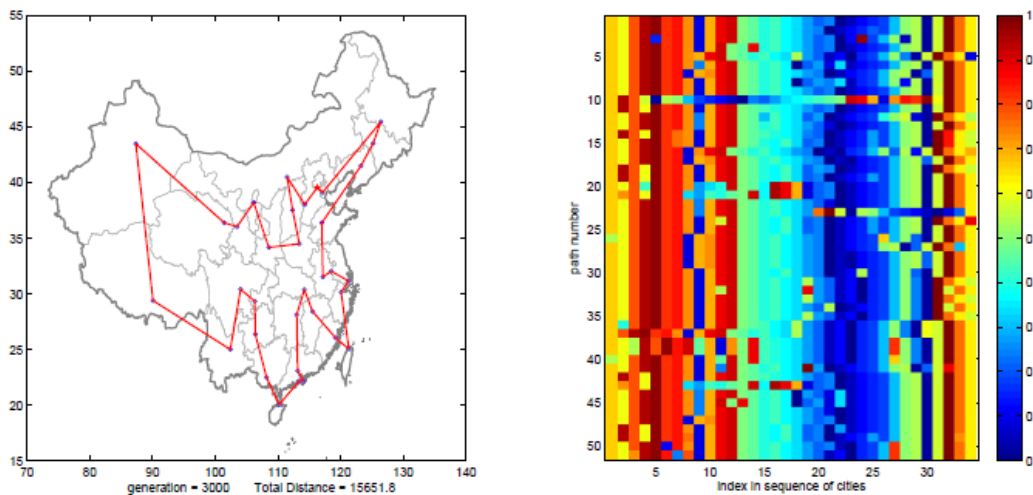


图 4-42

```

1 popSize = 50; % 种群规模
2 max_generation = 3000; % 初始化最大种群代数
3 Pmutation = 0.16; % 变异概率
4 for i = 1: popSize % 初始化种群
5     pop(i,:) = randperm(numberofcities);
6 end
7 for generation = 1: max_generation
8     fitness = 1/ totaldistance(pop ,dis); % 计算距离( 适应度)
9     [maxfit , bestID] = max(fitness);
10    bestPop = pop(bestID, :); % 找出精英
11    pop = select(pop,fitness); % 选择操作
12    pop = crossover(pop); % 交叉操作
13    pop = mutate(pop,Pmutation); % 变异操作
14    pop = [bestPop; pop]; % 精英保护
15 end
16
17 popDist = total_distance(pop ,dis); % 计算距离 (适应度)
18 [minDist , index] = min(popDist);
19 optRoute = pop(index ,:); % 找出最短距离

```

以上程序中第 10 行找出当前最好的解，并通过第 14 行将这个最好的解不做任何改变操作直接放到下一代的种群中，这样做是为了使得下一代的种群的最优解至少不会比上一代差。此外，上述主程序中还调用了函数 `select`，`crossover` 和 `mutate`，下面我们逐一介绍。

函数 `select` 是用来进行选择操作，在函数 `select` 中，我们提供两种选择算子：轮盘赌和两两竞争。函数 `select` 的代码如下。

```

1 function popselected = select(pop , fitness , nselected , method)
2 popSize = size(pop ,1);
3
4 switch method
5     case 'roulette ' % 轮盘赌
6         p=fitness/sum(fitness); % 选中概率
7         cump=cumsum(p); % 概率累加
8         % 利用插值: yi = 线性插值(x, y, xi)
9         I = interp1 ([0 ...cump ],1:( popSize +1),rand(1,
nselected),'linear ');
10        I = floor(I);
11
12        case 'competition ' % 两两竞争
13            i1 = randsample(popSize , nselected);
14            i2 = randsample(popSize , nselected);
15            I = i1.*( fitness(i1)-fitness(i2) ) + ...
16                i2.*( fitness(i1)< fitness(i2) );
17 end
18
19 popselected=pop(I,:);

```

以上代码中第6行得到规一化的适应度。在轮盘赌中,各染色体规一化的适应度即为其被选中的概率。第7行是对选中概率进行累加。如果种群的规模为5,种群中各染色体规一化的适应度为 $p=[0.16, 0.39, 0.11, 0.14, 0.20]$ ,则经过累加得到 $cump=[0.16, 0.55, 0.66, 0.8, 1.0]$ 。则可以通过生成随机来决定选择哪些染色体,若生成的随机 $rand \in [0, 0.16]$ ,则第一个染色体被选中;若生成的随机 $rand \in [0.66, 0.8]$ ,则第四个染色体被选中。第9行利用插值运算实现轮盘赌。以上程序中还提供了两两竞争的选择算子,这里不再详述。

函数 `crossover` 是用来进行交叉操作,由于需要消除交叉中出现的城市编号重得,交叉操作的代码稍微有点复杂。函数 `crossover` 的代码如下。

```

1 function children = crossover(parents)
2 [popSize , numberofcities] = size(parents);
3 children = parents; % 初始化子代
4
5 for i = 1:2: popSize
6     parent1 = parents(i+0,:); child1 = parent1;
7     parent2 = parents(i+1,:); child2 = parent2;
8     InsertPoints = sort(randsample(numberofcities , 2)); % 交叉点
9     for j = InsertPoints (1):InsertPoints (2)
10         if parent1(j) /= parent2(j) % 如果对应位置不重复

```







```

11     child1(child1 == parent2(j)) = child1(j);
12     child1(j) = parent2(j);
13
14     child2(child2 == parent1(j)) = child2(j);
15     child2(j) = parent1(j);
16     end
17 end
18 children(i+0,:)=child1; children(i+1,:)=child2;
19 end

```

以上程序中第 9 行中生成两个随机的交叉点。第 10 行至第 17 行的循环对两交叉点间的城市编号依次进行互换和消重。对于子代 1，由第 11 行消重，第 12 行互换以达到交叉的目的。第 11 行通过 `child1==parent2(j)` 找出交叉后会出现重复的位置，并替换为 `child1(j)`，那么第 12 行交叉后，`child1` 就不会出现重复的城市编码，对于子代 2，类似。

函数 `mutation` 是用来进行变异操作，在函数 `crossover` 提供了三种变异算子：“对调”，“滑移”和“逆序”。每一次变异都从三种变异算子中选择一种，并且主要以“逆序”为主。函数 `crossover` 的代码如下。

```

1 function children = mutation(parents , probmutation)
2 [popSize , numberofcities] = size(parents);
3 children = parents; % 初始化子代
4 for k=1: popSize
5     if rand < probmutation % 以一定概率变异
6         InsertPoints = randsample(numberofcities , 2);
7         I = min(InsertPoints); J = max(InsertPoints); % 两交叉点: I<J
8         switch randsample (6, 1) % 通过随机数判断使用哪种算子
9             case 1 % 对调
10                 children(k,[I J]) = parents(k,[J I]);
11             case 2 % 滑移
12                 children(k,[I:J]) = parents(k,[I+1:J I]);
13             otherwise % 逆序
14                 children(k,[I:J]) = parents(k,[J:-1:I]);
15         end
16     end
17 end

```

以上程序通过 `switch` 语句来选择不同的变异算子。第 8 行中 `randsample(6, 1)` 从  $[1, 2, \dots, 6]$  中随机抽取出的一个数字为 1 则采用“对调”算子；如果随机抽取出的数字为 2，则采用“滑移”算子；否则当随机抽取出的数字在  $[3, 4, 5, 6]$  中，则采用“逆序”算子。



## 第5章 赛题解析

### 5.1 2013 CUMCM-A

#### 5.1.1 问题综述

本案例讨论了车道被占用对城市道路通行能力的影响。车道被占用是指因交通事故、路边停车、占道施工等因素，导致车道或道路横断面通行能力在单位时间内降低的现象。一条车道被占用，也可能降低路段所有车道的通行能力。附件中视频 1 和视频 2 的两个交通事故处于同一路段的同一横断面，且完全占用两条车道。案例中主要讨论并建立模型，解决了如下 4 个问题。

问题 1：根据视频 1（附件 1），描述视频中交通事故发生至撤离期间，事故所处横断面实际通行能力的变化过程。

问题 2：根据问题 1 所得结论，结合视频 2（附件 2），分析说明同一横断面交通事故所占车道不同对该横断面实际通行能力影响的差异。

问题 3：构建数学模型，分析视频 1（附件 1）中交通事故所影响的路段车辆排队长度与事故横断面实际通行能力、事故持续时间、路段上游车流量间的关系。

问题 4：假如视频 1（附件 1）中的交通事故所处横断面距离上游路口变为 140 米，路段下游方向需求不变，路段上游车流量为 1500pcu/h，事故发生时车辆初始排队长度为零，且事故持续不撤离。请估算，从事故发生开始，经过多长时间，车辆排队长度将到达上游路口。

#### 5.1.2 分析与建模及求解

##### 一、问题的分析

##### （一）问题 1 的分析

案例中，问题 1 主要考察对抽象数据的处理能力和统计数据的分析能力。在数据处理方面，为分析事故所处横断面实际通行能力的变化，首先对视频 1 中交通事故发生至撤离期间的录像进行处理，数出以 30s 为时间间隔，红绿灯交替 1min 为一个周期的单位时间内横断面通过的车量个数，计算实际通行能力。由于视频中有多处出现片段的间断，影响计





数, 为了得到完整的数据, 同时考虑到经过横断面的车辆数为随机数这一情况, 我们选用 Gibbs 抽样仿真方法, 对缺失的数据进行填补。用填补后的数据绘制折线图, 描述交通事故发生至撤离期间, 事故所处横断面实际通行能力的变化过程。

## （二）问题 2 的分析

案例中问题 2 主要考察相关的统计知识, 包括对非参检验的应用和统计数据和分析能力。同时也要求对整个题目能有整体的把握, 理清题目的来龙去脉, 理解题目的含义以及题目想要得到的结果。

根据视频 1 的处理方法同样统计出视频 2 中的各项数据, 然后分别从 60s 周期、上游路口为红灯和上游路口为绿灯时候的通行能力, 将两种不同横断面的情况进行对比, 利用非参数检验中两独立样本的曼-惠特尼 U 检验对两组通行能力差异进行对比, 并根据结果进行相应的分析。

## （三）问题 3 的分析

问题 3 是整个题目问题中的核心, 需要建立一个或多个模型求解。问题 3 要求分析视频 1 中交通事故所影响的路段车辆排队长度与事故横断面实际通行能力、事故持续时间、路段上游车流量间的关系。也就是说我们需要建立一个模型把排队长度和事故横断面实际通行能力、事故持续时间、路段上游车流量间这几个因素联系起来, 再结合图形分析它们之间的关系。

当发生交通事故之后, 没有大量的车流入, 原因当时是红灯, 半分钟之后, 有大量的车辆流入, 原因是红灯变为了绿灯, 视频中参与到排队的流入车辆包括了上游路口进入的车辆和小区出口出来的车辆, 没有参加排队的车辆包括了从小区入口去小区的车辆和从事故横断面离开的车辆, 经过检验, 发现车辆流入是一个服从泊松分布的随机过程, 且三个车道被堵住了两个, 所以考虑建立单服务窗口排队论模型, 然而这与通常的稳态排队论问题不同, 我们决定重新审视排队论问题中的生灭过程, 求解非稳态排队论通解, 得出路段车辆排队长度与事故横断面实际通行能力、事故持续时间、路段上游车流量间的关系。

再考虑到流入车辆, 流出车辆和堵塞在排队的车辆的守恒关系, 我们选择建立简单的差分方程模型, 再考虑到红绿灯对流入量的影响, 我们选择建立分段差分方程来建立路段车辆排队长度与事故横断面实际通行能力、事故持续时间、路段上游车流量间的关系。

其中通过对视频 1 的仔细观察, 我们发现当靠小区出口的道路排队队长超过小区路口位置的时候, 小区出来的车与进入道路的车辆会增加队长, 但在, 小区路口等待的车辆却不增加队长, 针对这一情况完善我们的模型。

## （四）问题 4 的分析

问题 4 求对从事故发生开始, 经过多长时间车辆排队长度将到达上游路口。实际上是对问题 3 建立的模型的实际应用, 也是对问题 3 中建立模型的检验。

考虑到事故发生在距离路口 140m 处，不再有小区 1 车辆进出参加排队，但是却有车辆进出小区 2，于是改变问题三中两种模型的参数，针对上游车流量为 1500pcu/h，用排队论模型进行求解，再考虑到红绿灯对队长的影响，分别假设事故发生时间是在红灯开始时刻和绿灯开始时刻，用分段差分方程求解。

二、模型的假设

- (1) 视频中所统计数据真实可靠。
- (2) 排队所占车道车辆数与对应车道行驶方向车辆数成正比，即：车道一车辆数：车道二车辆数：车道三车辆数=0.21：0.44：0.35。
- (3) 除事故车辆外的其他车辆严格遵守交通规则，红灯停，绿灯行。
- (4) 车辆到达率与正在排队车辆数量无关，无论有多少车在排队，车辆到达率不变。
- (5) 车辆来源是无限的。
- (6) 堵车期间该路段没有其他交通事故发生。
- (7) 在堵车状况下相邻两辆车车头之间间距为 7m。

三、模型的建立及问题的求解

接下来对案例中不同问题进行分别地解答：在前两问中主要进行数据的处理和分析，运用统计知识解决问题 1 和问题 2，并在第三问中建立本案例最关键的两个模型，即非稳态排队论模型和分段差分模型，来解决问题 3 中的问题。

(一) 问题 1 的求解

1. 数据的处理与分析

- (1) 对视频 1 中事故所处横断面通过车辆进行计数。
- 对经过事故所处横断面的车辆进行计数，按照规定，只考虑四轮及以上机动车、电瓶车的交通流量，且换算成标准车当量数，根据《公路工程技术标准》(JTG B01—2003)规定的换算标准，对视频中符合要求的车辆数据进行换算，其折算规则如表 5-1 所列。

表 5-1

车 型 车	载荷及功率	折算系数
小客车	额定座位≤19 座	1.0
大客车	额定座位≥19 座	1.5
小型货车	载重量≤2 吨	1.0
大型货车	载重量≥2 吨	1.5

从视频可知事故发生时间为 16：42：32 附近，所以我们以 30s 为时间间隔，以 16：42：30 为时间起点，开始计数。车辆计数及当量转换数据结果如表 5-2 所列。



表 5-2

时间	小车	大车	当量转换	通行能力	时间	小车	大车	当量转换	通行能力
42:30	7	2	10	1200	43:00	8	1	9.5	1140
43:30	9	1	10.5	1260	44:00	8	1	9.5	1140
44:30	8	0	8	960	45:00	8	0	8	960
45:30	7	1	8.5	1020	46:00	7	1	8.5	1020
46:30	9	0	9	1080	47:00	7	0	7	840
47:30	7	1	8.5	1020	48:00	11	0	11	1320
48:30	10	0	10	1200	49:00	9	0	9	1080
49:30	9	0	9	1080	50:00	7	1	8.5	1020
50:30	10	0	10	1200	51:00	9	0	9	1080
51:30	8	0	8	960	52:00	9	1	10.5	1260
52:30	8	1	9.5	1140	53:00	8	0	8	960
53:30	9	0	9	1080	54:00	9	0	9	1080
54:30	8	1	9.5	1140	55:00	9	0	9	1080
55:30	10	0	10	1200	56:00	8	0	8	960
56:30	9	0	9	1080	57:00	7	1.5	8.5	1020
57:30	9	0	9	1080	58:00	8	0	8	960
58:30	7	1	8.5	1020	59:00	7	1	8.5	1020

注：粗体部分数据为 Gibbs 抽样仿真方法预测数据。

## （2）用 Gibbs 抽样仿真方法进行处理。

对空缺数据的处理方法有很多，案例中选择 Gibbs 抽样方法对空缺数据进行预测的原因是 Gibbs 抽样表现为一个 Markov 链形式的 Monte Carlo 方法，其良好的性质可用于许多随机系统的分析、多元分布的随机数产生。这样产生的数据可以继承原数据良好的统计性质。

上游路口是红灯还是绿灯对交通事故横断面实际通行能力会造成一定的影响，所以红灯和绿灯作为两种情况分开考虑，计算不同情况下通行能力出现的条件概率。对视频中缺失片段数据进行补全，将已知片段按周期分开得到事故发生至撤离期间红绿灯交替对应车辆当量如表 5-3 所列。

表 5-3

红灯	10	10.5	8	7	9	8.5	10	10	8	9.5	9.5	9.5
绿灯	9.5	9.5	8	8.5	7	11	9	9	10.5	8	9	9

根据表 5-2 中统计的数据（通行能力已知部分）可得，横断面可能出现的车辆数集合为 (7,8,8.5,9,9.5,10,10.5,11)，对应数字 (1,2,3,4,5,6,7,8) 分别表示这 8 个数。运用 Gibbs 抽样填充缺失数据只要掌握缺失数据的属性与其他属性之间的条件分布，就能够利用这些分



布产生数据，因此我们对数据进行整理得出条件概率分布表如表 5-4 所列。

表 5-4

标号	1	2	3	4	5	6	7	8	
车辆当量 交通灯	7	8	8.5	9	9.5	10	10.5	11	合计
红灯	$\frac{1}{12}$	$\frac{2}{12}$	$\frac{1}{12}$	$\frac{1}{12}$	$\frac{3}{12}$	$\frac{3}{12}$	$\frac{1}{12}$	0	100%
绿灯	$\frac{1}{12}$	$\frac{2}{12}$	$\frac{1}{12}$	$\frac{4}{12}$	$\frac{2}{12}$	0	$\frac{1}{12}$	$\frac{1}{12}$	100%

Excel 来实现 Gibbs 抽样仿真。首先使用 rand()命令产生随机数数列。在 Excel 中，rand()产生的是 0 到 1 之间均匀分布的随机数，因而所产生的随机数小于 0.6 的概率就是 0.6。使用的 Excel 命令见附件 1。

填补未知数据结果如表 5-5 所列。

表 5-5

时 间	车辆当量	通行能力	时 间	车辆当量	通行能力	时 间	车辆当量	通行能力
56:00	8	960	56:30	9	1080	57:00	8.5	1020
57:30	9	1080	58:00	8	960	58:30	8.5	1020
59:00	8.5	1020	49:30	9	1080	53:30	9	1080

将补全后的数据，运用 Excel 画出折线图如图 5-1 所示。

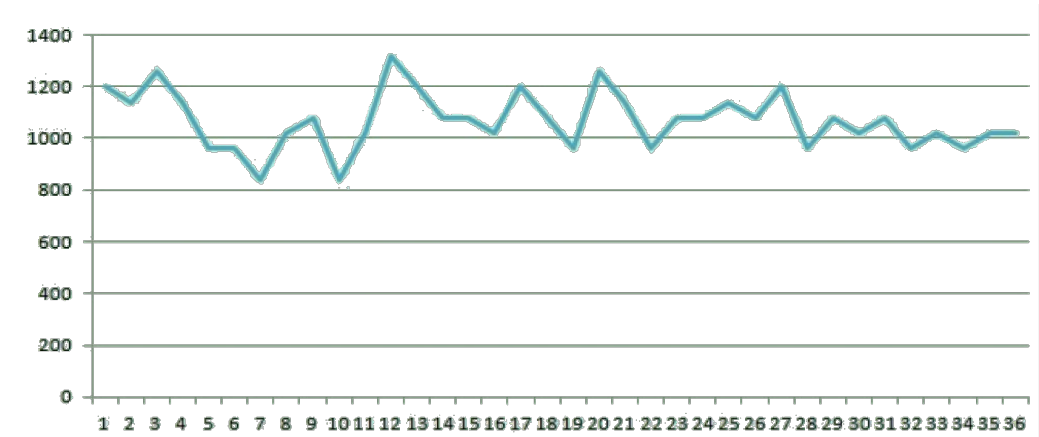


图 5-1

由于数据呈现不平稳波动，因此我们考虑运用 eviews 的 ARIMA 模型对数据进行拟合，我们定义开始时间，做出自相关图如图 5-2 所示。



Date: 09/16/13 Time: 08:46  
Sample: 1 36  
Included observations: 36

Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 0.204	0.204	1.6266	0.202
		2 -0.047	-0.093	1.7158	0.424
		3 0.060	0.094	1.8633	0.601
		4 -0.041	-0.084	1.9350	0.748
		5 -0.160	-0.129	3.0670	0.690
		6 -0.174	-0.135	4.4466	0.616
		7 -0.308	-0.286	8.9282	0.258
		8 0.027	0.159	8.9649	0.345
		9 0.089	0.008	9.3650	0.404
		10 -0.117	-0.129	10.087	0.433
		11 0.035	0.034	10.153	0.517
		12 0.110	-0.041	10.839	0.543
		13 -0.061	-0.118	11.061	0.606
		14 0.014	-0.000	11.073	0.680
		15 -0.050	-0.063	11.236	0.736
		16 -0.067	-0.025	11.543	0.775

图 5-2

发现自相关图不稳定，因此对其进行一阶差分后得到自相关图如图 5-3 所示，并进行单位根 ADF 检验。

Date: 09/16/13 Time: 08:37  
Sample: 1 36  
Included observations: 35

Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 -0.342	-0.342	4.4545	0.035
		2 -0.262	-0.429	7.1528	0.028
		3 0.161	-0.158	8.2062	0.042
		4 0.043	-0.077	8.2840	0.082
		5 -0.059	-0.036	8.4333	0.134
		6 0.099	0.124	8.8673	0.181
		7 -0.335	-0.359	14.071	0.050
		8 0.156	-0.156	15.234	0.055
		9 0.241	0.053	18.117	0.034
		10 -0.276	-0.128	22.063	0.015
		11 -0.007	-0.037	22.065	0.024
		12 0.177	-0.006	23.834	0.021
		13 -0.136	-0.106	24.920	0.024
		14 -0.080	-0.034	25.312	0.032
		15 -0.005	-0.048	25.314	0.046
		16 -0.087	-0.030	25.827	0.057

Null Hypothesis: DX has a unit root  
Exogenous: Constant  
Lag Length: 1 (Automatic - based on SIC, maxlag=8)

	t-Statistic	Prob.*
Augmented Dickey-Fuller test statistic	-7.145431	0.0000
Test critical values:		
1% level	-3.646342	
5% level	-2.954021	
10% level	-2.615817	

\*MacKinnon (1996) one-sided p-values.

Augmented Dickey-Fuller Test Equation  
Dependent Variable: D(DX)  
Method: Least Squares  
Date: 09/16/13 Time: 09:52  
Sample (adjusted): 4 36  
Included observations: 33 after adjustments

Variable	Coefficient	Std. Error	t-Statistic	Prob.
DX(-1)	-1.910866	0.267425	-7.145431	0.0000
D(DX(-1))	0.429937	0.163176	2.634799	0.0132

图 5-3

发现在显著性水平 0.05 下可以拒绝一个单位根的原假设，所以一阶差分后序列已经稳定。由于一阶差分自相关系数在 2 阶和 3 阶出落在 2 倍标准差边缘，因此考虑用 ma（2）进行尝试，最终选择拟合程度最好的模型为 ARMA(0, 2)。拟合 R 方为 0.411，拟合图形如图 5-4 所示。

因此拟合曲线为：

$$y_t = -2.0232 + \epsilon_t - 0.7243\epsilon_{t-1} - 0.2757\epsilon_{t-2}$$

Dependent Variable: D(X)  
Method: Least Squares  
Date: 09/16/13 Time: 08:13  
Sample (adjusted): 2 36  
Included observations: 35 after adjustments  
Convergence achieved after 18 iterations  
MA Backcast: 0 1

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	-2.023179	1.664548	-1.215452	0.2331
MA(1)	-0.724266	0.168687	-4.293551	0.0002
MA(2)	-0.275717	0.171623	-1.606527	0.1180
R-squared	0.411052	Mean dependent var	-5.142857	
Adjusted R-squared	0.374243	S.D. dependent var	139.8607	
S.E. of regression	110.6365	Akaike info criterion	12.33219	
Sum squared resid	391693.9	Schwarz criterion	12.46551	
Log likelihood	-212.8134	Hannan-Quinn criter.	12.37821	
F-statistic	11.16710	Durbin-Watson stat	2.097176	
Prob(F-statistic)	0.000210			
Inverted MA Roots	1.00	-0.28		

图 5-4

结合视频对图像进行分析，横断面的实际通行能力呈现上下波动的不稳定状态，但整体都在一条直线上上下波动。在第二个时间点下降是因为车祸发生的时间刚好是上一次绿灯通过大量车到达车祸截面，由于人们原本还在自己选择的车道上，但是当发现车祸后，均会转移到右转车道，因此右转车道在开始短时间内可以顺利通行，当其他车道挤过来时，会产生排队效应，降低该车道的实际通行能力，因此第二个时间点的实际通行能力会比第一个时间点低。加上上游路口红绿灯是以 60s 为周期，所以整个截面通行能力呈现上下波动的情况。

（二）问题 2 的求解

案例中问题 2 要求由于占用车道不同，而对横断面的道路通行能力的影响进行对比，本论文选择了非参检验来对两种情况进行差异分析。实际上是利用非参数检验中两独立样本的曼-惠特尼 U 检验对两组通行能力差异进行对比，并根据结果进行相应的分析。





## 1. 数据收集

视频 2 车辆计数及当量转换如表 5-6 所列。

表 5-6

时间 $t$	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
车流量 pcu/h	1260	1620	1500	1020	1260	1440	1500	1080	1500	900	1560	720
时间 $t$	6.5	7	7.5	8	8.5	9	9.5	10	10.5	11	11.5	12
车流量 pcu/h	1680	960	1380	1320	1620	1380	1260	720	1140	960	1440	1140
时间 $t$	12.5	13	13.5	14	14.5	15	15.5	16	16.5	17	17.5	18
车流量 pcu/h	1140	360	1500	1260	1140	1200	1440	1380	1200	1200	1260	1080
时间 $t$	18.5	19	19.5	20	20.5	21	21.5	22	22.5	23	23.5	24
车流量 pcu/h	1260	1140	960	1320	1260	1440	1020	1500	1200	1320	1200	1200
时间 $t$	24.5	25	25.5	26	26.5	27	27.5	28	28.5	29	29.5	
车流量 pcu/h	1080	1080	840	1260	1260	1140	1080	1140	1380	1320	1260	

## 2. 问题解答

首先以 60s 为周期对视频 1 和 2 横断面实际通行能力的差异进行比较, 利用 SPSS 进行曼-惠特尼 U 检验操作得到如图 5-5 所示结果。

Mann-Whitney U	508
Wilcoxon W	1174
Z	-4.272
渐近显著性(双侧)	0.0000

变量	N	秩均值	秩和
1	36	32.61	1174
2	59	57.39	3386
总数	95		

图 5-5

由表格可知, 检验  $p$  值小于 0.05, 因此我们要拒绝原假设, 且视频 2 的秩均值大于视频 1, 即可以认为以 60s 为周期时同一横断面交通事故所占车道不同对该横断面实际通行能力的影响有显著差异, 且视频 2 的实际通行能力比视频 1 大。

然后用同样的方法, 分别对上游路口为红灯和上游路口为绿灯时的实际通行能力差异进行比较, 同样利用 SPSS 进行曼-惠特尼 U 检验操作得到红灯[图 5-6 (a) 所示]和绿灯[图 5-6 (b) 所示]的结果。



Mann-Whitney U	97	Mann-Whitney U	151
Wilcoxon W	268	Wilcoxon W	322
Z	-3.712	Z	-2.421
渐近显著性(双侧)	0.0000	渐近显著性(双侧)	0.01547

变量	N	秩均值	秩和
1	18	14.89	268
2	30	30.27	908
总数	48		

(a)

变量	N	秩均值	秩和
1	18	17.89	322
2	29	27.79	806
总数	47		

(b)

图 5-6

由图 5-6 (a) 可知，检验  $p$  值小于 0.05，因此我们要拒绝原假设，且视频 2 的秩均值大于视频 1。即可以认为在红灯时同一横断面交通事故所占车道不同对该横断面实际通行能力的影响有显著差异。同样的，由图 5-6 (b) 可知，检验  $p$  值小于 0.05，因此我们要拒绝原假设，且视频 2 的秩均值大于视频 1，即可以认为在绿灯时同一横断面交通事故所占车道不同对该横断面实际通行能力的影响有显著差异。且视频 2 的实际通行能力比视频 1 大。

综合上述三个检验，我们可以认为同一横断面交通事故所占车道不同对该横断面实际通行能力影响有显著差异，且这里视频 2 的实际通行能力比视频 1 大。我们画出折线图进行比较如图 5-7 所示。

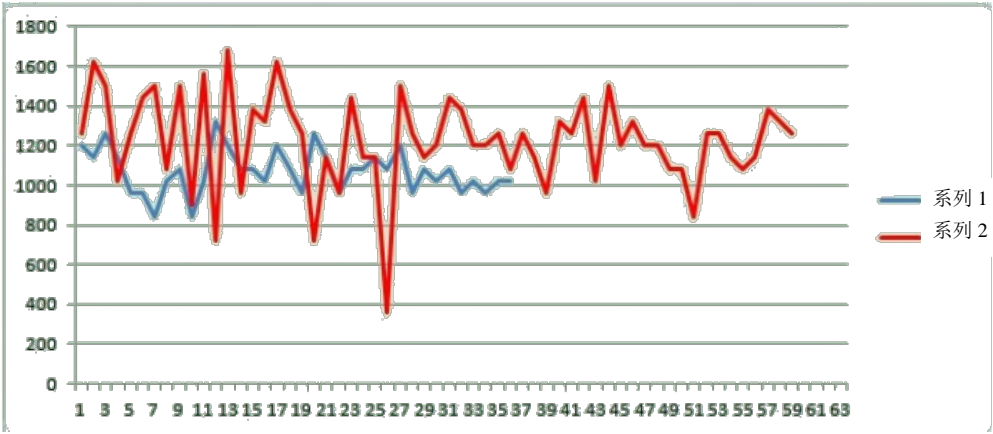


图 5-7

我们以视频 1 为比较基准，分别计算 60s 为周期，绿灯和红灯时候视频 2 中截面通行能力的变化率，其中通行能力取各个情况下的均值作为比较。计算可得 60s 内视频 2 相对于视频 1 的通行能力变化率为 14.455%。绿灯时候的变化率为 18.344%。红灯时候的变化率为 10.345%。

根据路面的实际情况造成这种结果的原因有如下两个方面：

① 由于到下游路口直行和左转流量比例总共为 79%。当人们从上游路口进入时，会按照自己意愿选择车道，因此大部分人会在直行和左转车道上，所以当这个两条道路被堵



时,道路上车为了通行必须拐到右转车道,这种类似插队的行为效率较低,造成时间浪费,而相反,如果是右转车道和直行车道被堵时,这两个车道的流量比例总共为 65%,那么时间浪费的次数会相对第一种情况较少,因此通行能力会更好。所以当上游路口为绿灯的时候,车流量较大,直行道和左转车道的数量会较多,因此相对于红灯时造成的排队效应会更大,时间浪费更多,所以绿灯变化率会比红灯时候大。

② 由于小区路口在右转车道边上,如果左转和直行车道被堵,那么当右转车道队长到小区路口时,小区车的出入就会受到影响,同时也会影响右边车道的流通,而如果当右边车道和直行车道被堵时,即使右边车道队长到小区门口,对左边车道通行的影响也不会太大。

### (三) 问题 3 的解答

这也是案例中最关键的部分,在问题 3 的解答这一节中,本论文首先针对题目要求建立模型,为了更好地分析和解决问题,论文中用两种不同的思路分别建立了分段差分模型和非稳态排队论模型。下面分别对这两种模型进行详细介绍。

#### 1. 数据处理

对于问题三,我们考虑到不仅是上游路口有车流入,两个小区也有车流入和流出,因此我们定义一个净流入的车流量为上游路口车流入量加上从小区流入路段的车流量再减去流入小区的车流量,将这个净车流量作为到达的人数,并利用 SPSS 进行泊松分布的检验得到结果如图 5-8 所示。

单样本 Kolmogorov-Smirnov 检验 3

		VAR00003
N		19
Poisson 参数 <sup>a,b</sup> 均值		1.894737
最极端差别	绝对值	.249
	正	.249
	负	-.150
Kolmogorov-Smirnov Z		1.085
渐近显著性(双侧)		.190

- ① 检验分布为 Poisson 分布。  
② 根据数据计算得到。

图 5-8

由检验结果可知, $p$  值为 0.19,所以在置信水平 0.05 下我们可以接受原假设,即净流入车流量与泊松分布无显著差异,即满足泊松分布。

当左转车道的队列长度排到第一个小区口时,这个小区口的车就无法进入到车道上,此时能流入到车道的车流量只有上游路口的流入加上第一个小区的净流出,同样我们将此时的净车流量作为到达人数,并利用 SPSS 进行泊松分布的检验得到结果如图 5-9 所示。



单样本 Kolmogorov-Smirnov 检验 3			VAR00005
N			19
Poisson 参数 <sup>a,b</sup>	均值		1.894737
最极端差别	绝对值		.196
	正		.196
	负		-.150
Kolmogorov-Smirnov Z			.856
渐近显著性(双侧)			.457

- ① 检验分布为 Poisson 分布。
- ② 根据数据计算得到。

图 5-9

由检验结果可知， $p$  值为 0.457，所以在置信水平 0.05 下我们可以接受原假设，即净流入车流量与泊松分布无显著差异，即满足泊松分布。

考虑到我们要进行红绿灯的分段计算，因此我们分别对红灯和绿灯时的净车流量进行泊松检验分别得到如图 5-10 所示的结果。

单样本 Kolmogorov-Smirnov 检验 3			VAR00007
N			5
Poisson 参数 <sup>a,b</sup> 均值			3.600000
最极端差别	绝对值		.306
	正		.274
	负		-.306
Kolmogorov-Smirnov Z			.685
渐近显著性(双侧)			.736

单样本 Kolmogorov-Smirnov 检验 3			VAR00009
N			14
Poisson 参数 <sup>a,b</sup> 均值			1.285714
最极端差别	绝对值		.276
	正		.154
	负		-.276
Kolmogorov-Smirnov Z			1.034
渐近显著性(双侧)			.235

- ①检验分布为 Poisson 分布。
- ②根据数据计算得到。

(a)

- ①检验分布为 Poisson 分布。
- ②根据数据计算得到。

(b)

图 5-10

由检验结果可知， $p$  值分别为 0.736 和 0.235，所以在置信水平 0.05 下我们可以接受原假设，即红灯和绿灯时候的净流入车流量与泊松分布无显著差异，即满足泊松分布。

2. 问题 3 模型的建立

(1) 第一种模型：非稳态排队论模型

首先，建立堵车过程的微分方程组。由堵车过程的前提条件可知，状态  $N(t + \Delta t) = j$ ，只能由三种状态转移而来，即  $N(t) = j; N(t) = j - 1; N(t) = j - 1$ 。由全概率公式，得

$$P_j(t + \Delta t) = \lambda_{j-1}\Delta t P_{j-1}(t) + [1 - (\lambda_j + \mu_j)\Delta t] P_j(t) + \mu_j \Delta t P_{j+1}(t) + o(\Delta t)$$



所以

$$\lim_{\Delta t \rightarrow \infty} \frac{P_j(t + \Delta t) - P_j(t)}{\Delta t} = \lambda_{j-1}P_{j-1}(t) - (\lambda_j + \mu_j)P_j(t) + \mu_{j+1}P_{j+1}(t)$$

即

$$P_j'(t) = \lambda_{j-1}P_{j-1}(t) - (\lambda_j + \mu_j)P_j(t) + \mu_{j+1}P_{j+1}(t)$$

当  $j=0$  时

$$P_0'(t) = \mu_1P_1(t) - \lambda_0P_0(t)$$

考虑道路车辆容纳量有限, 最大容纳量  $n$ , 则

$$P_n'(t) = \lambda P_{n-1}(t) - \mu P_n(t)$$

上述公式 (1) (2) (3) 组成堵车过程的微分方程组

$$\begin{cases} P_0'(t) = \mu P_1(t) - \lambda P_0(t) \\ P_j'(t) = \lambda_{j-1}P_{j-1}(t) - (\lambda_j + \mu_j)P_j(t) + \mu_{j+1}P_{j+1}(t) \quad (j=1, 2, \dots, n-1) \\ P_n'(t) = \lambda P_{n-1}(t) - \mu P_n(t) \end{cases}$$

考虑到交通堵塞不是一般的排队论问题, 来的车往往比离开车多, 造成了交通堵塞, 所以, 交通堵塞问题中的到达率  $\lambda$ , 离开率  $\mu$  的比值大于  $\frac{\lambda}{\mu} > 1$ , 这样一来, 上面的常微分方程组中  $P_j$  是随着时间变化而变化的因变量, 所以不能只考虑稳态方程的解, 下面考虑此方程组的通解:

方程组即为:  $\frac{d\vec{P}}{dt} = A\vec{P}$

其中:

$$A = \begin{pmatrix} -\lambda & \mu & 0 & \dots & \dots \\ \lambda & -\lambda - \mu & \mu & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \lambda & -\lambda - \mu & \mu \\ \dots & \dots & 0 & \lambda & -\mu \end{pmatrix}$$

$$\vec{P} = \begin{pmatrix} P_0(t) \\ P_1(t) \\ \dots \\ P_n(t) \end{pmatrix}$$

我们来求解形如  $\vec{P} = \vec{r}e^{xt}$  的解, 其中  $x$  为矩阵  $\mathbf{A}$  的特征值,  $r$  为  $x$  对应的特征向量;  
则特征方程  $|xI - \mathbf{A}| = 0$  的解如下:

$$|xI - \mathbf{A}| = \begin{vmatrix} x + \lambda & -\mu & & & & \\ -\lambda & x + \lambda + \mu & -\mu & & & \\ & -\lambda & x + \lambda + \mu & -\mu & & \\ \dots & \dots & \dots & \dots & \dots & \dots \\ & & & -\lambda & x + \lambda + \mu & -\mu \\ & & & & -\lambda & x + \mu \end{vmatrix}$$

$$= \mathbf{A}_{n+1} - \mu \mathbf{A}_n - \lambda \mathbf{A}_n + \lambda \mu \mathbf{A}_{n-1}$$

其中:

$$\mathbf{A}_n = \begin{vmatrix} x + \lambda & -\mu & & & & \\ -\lambda & x + \lambda + \mu & -\mu & & & \\ & -\lambda & x + \lambda + \mu & -\mu & & \\ \dots & \dots & \dots & \dots & \dots & \dots \\ & & & -\lambda & x + \lambda + \mu & -\mu \\ & & & & -\lambda & x + \mu \end{vmatrix}$$

是一个  $n \times n$  的行列式。

所以,  $\mathbf{A}_{n+1} = (x + \lambda + \mu)\mathbf{A}_n - \lambda \mu \mathbf{A}_{n-1}$

$$|xI - \mathbf{A}| = (x + \lambda + \mu)\mathbf{A}_n - \lambda \mu \mathbf{A}_{n-1} - \mu \mathbf{A}_n - \lambda \mathbf{A}_n + \lambda \mu \mathbf{A}_{n-1} = x \mathbf{A}_n$$

显然  $x=0$  是其中一个特征值, 由  $|\mathbf{A}_n| = 0$  可以求得其他特征值

$$\mathbf{A}_n = \frac{(x + \lambda + \mu + \sqrt{-\Delta i})^{n+1} - (x + \lambda + \mu - \sqrt{-\Delta i})^{n+1}}{2^{n+1} \sqrt{-\Delta i}}$$

其中  $\Delta = \sqrt{a^2 - 4bc}$ ,  $a = x + \lambda + \mu$ ,  $b = -\mu$ ,  $c = -\lambda$

设复角  $\theta$  为  $a + \sqrt{4bc - a^2 i}$  的复角。

由  $\mathbf{A}_n = 0$  得

$$(\cos \theta + i \sin \theta)^{n+1} = [\cos(2\pi - \theta) + i \sin(2\pi - \theta)]^{n+1}$$

即:

$$(\cos \theta + i \sin \theta)^{n+1} = (\cos \theta - i \sin \theta)^{n+1}$$

进而

$$\cos[(n+1)\theta] + i \sin[(n+1)\theta] = \cos[(n+1)\theta] - i \sin[(n+1)\theta]$$

所以  $\sin[(n+1)\theta] = 0$ , 则  $\theta = \frac{k\pi}{n+1}$ ,  $k=1, 2, \dots, n$





$$\text{由 } \cos\left(\frac{k\pi}{n+1}\right) = \frac{a}{\sqrt{4bc}} = \frac{x+\lambda+\mu}{\sqrt{4\lambda\mu}}$$

得

$$x_k = 2\sqrt{\lambda\mu} \cos \frac{k\pi}{n+1} - \lambda - \mu, k=1, 2, \dots, n$$

所以  $A$  的  $n+1$  个特征值  $x_0, x_1, \dots, x_n$  和对应的  $n+1$  特征向量  $\vec{r}_0, \vec{r}_1, \dots, \vec{r}_n$ ; 其中:

$$\begin{cases} x_0 = 0 \\ x_k = 2\sqrt{\lambda\mu} \cos \frac{k\pi}{n+1} - \lambda - \mu \quad (k > 0) \end{cases}$$

由均值不等式, 当  $\lambda \neq \mu$  时,  $\lambda + \mu > 2\sqrt{\lambda\mu}$ , 所以  $x_k < 0$

用 MatLab 可求得特征向量

$$\vec{r}_j = \begin{pmatrix} r_{(j,1)} \\ r_{(j,2)} \\ \dots \\ r_{(j,n)} \end{pmatrix}$$

所以方程的通解为:  $\vec{P} = \sum_{k=0}^n c_k e^{x_k t} \vec{r}_k$

初始状况下有  $k$  辆车已经在排队, 则初始条件  $P_k(0)=1, P_j(0)=0$ , ( $j \neq k$  时) 从而有

$$\begin{cases} 0 = c_0 r_{(0,0)} + c_1 r_{(1,0)} + \dots + c_n r_{(n,0)} & (1) \\ \vdots \\ 0 = c_0 r_{(0,k-1)} + c_1 r_{(1,k-1)} + \dots + c_n r_{(n,k-1)} & (k-1) \\ 1 = c_0 r_{(0,k)} + c_1 r_{(1,k)} + \dots + c_n r_{(n,k)} & (k) \\ 0 = c_0 r_{(0,k+1)} + c_1 r_{(1,k+1)} + \dots + c_n r_{(n,k+1)} & (k+1) \\ \vdots \\ 0 = c_0 r_{(0,n)} + c_1 r_{(1,n)} + \dots + c_n r_{(n,n)} & (n) \end{cases}$$

于是得到任意塞车时刻  $t$  时, 在排队的车辆的数量为  $j$  的概率为

$$P_j(t) = \sum_{k=0}^n c_k e^{x_k t} r(k, j)$$

平均队长 (辆)

$$m(t) = \sum_{i=0}^n i P_i(t) = \sum_{i=0}^n i \sum_{k=0}^n c_k e^{x_k t} r(k, i)$$

路段车辆排队长度 (m)



$$s(t) = l_0 m(t) \times \max\{P_i\} = 0.44 l_0 m(t)$$

考虑到当路段右转车道车辆排队长度超过小区路口的时候，下游小区车辆无法进出，所以车辆到达率：

$$\lambda = \begin{cases} \lambda_0 + a_1 - a_2 & (0.21 l_0 m(t) < 60) \\ \lambda_0 - a_2 & (0.21 l_0 m(t) \geq 60) \end{cases}$$

其中： $\lambda_0$  是单位时间上游路口流入车量， $a_1$  是单位时间小区路口流入车量， $a_2$  是单位时间驶入第二小区车量， $\mu$  为实际通行能力。

这时初始条件变为  $P_k(t_0) = 1, P_j(t_0) = 0$  ( $j \neq k$  时)，其他不变。

图 5-11 所示是用 MatLab 根据以上非稳态排队论模型做出的路段车辆排队长度与路段上游车流量的关系图，发现路段上游车流量越大，路段车辆排队长度越大，塞车越严重，直到赌满到路口为止。说明上游车流量大的路段需要对交通事故的发生做好防范，以免造成严重的交通堵塞，这一现象实际情况一致，很好的反应了实际问题，说明此模型具有很高的实用价值。

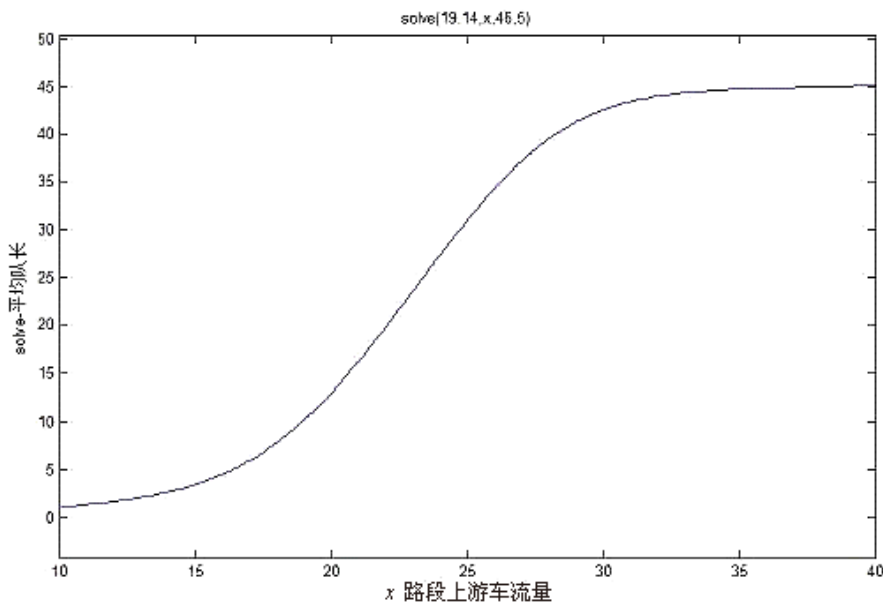


图 5-11

图 5-12 是用 MatLab 根据以上非稳态排队论模型做出的路段车辆排队长度与横断面实际通行能力的关系图，发现横断面实际通行能力越大，路段车辆排队长度越小，拥堵程度越轻。说明横断面实际通行能力差的地方需要多提醒人们注意交通安全，以免造成严重的交通堵塞，这一现象与实际情况一致，很好的反应了实际问题，说明此模型具有使用价值。

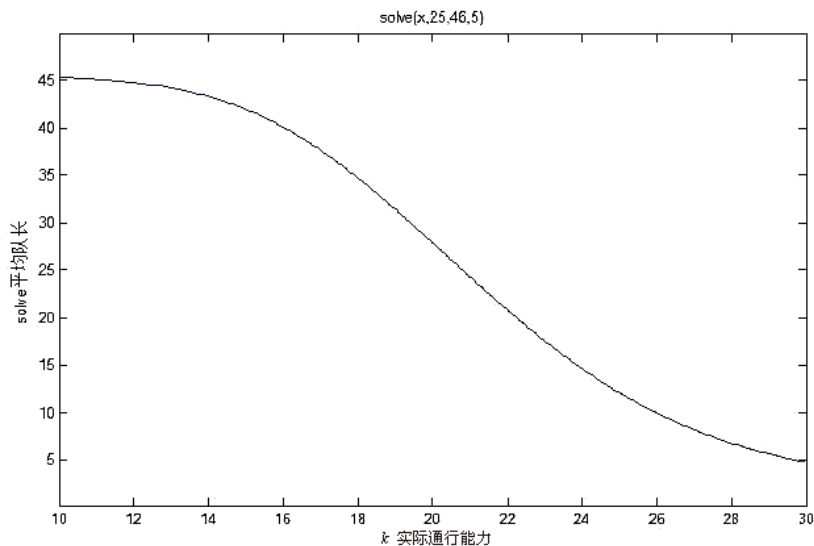


图 5-12

图 5-13 是用 MatLab 根据以上非稳态排队论模型做出的路段车辆排队长度与事故持续时间的关系图，发现事故持续时间越久，路段车辆排队长度越长，塞车越严重。说明发生交通事故之后，交警应该尽快解决交通问题，使得交通恢复正常运作，这一现象与实际情况一致，很好的反应了实际问题，说明此模型具有使用价值。

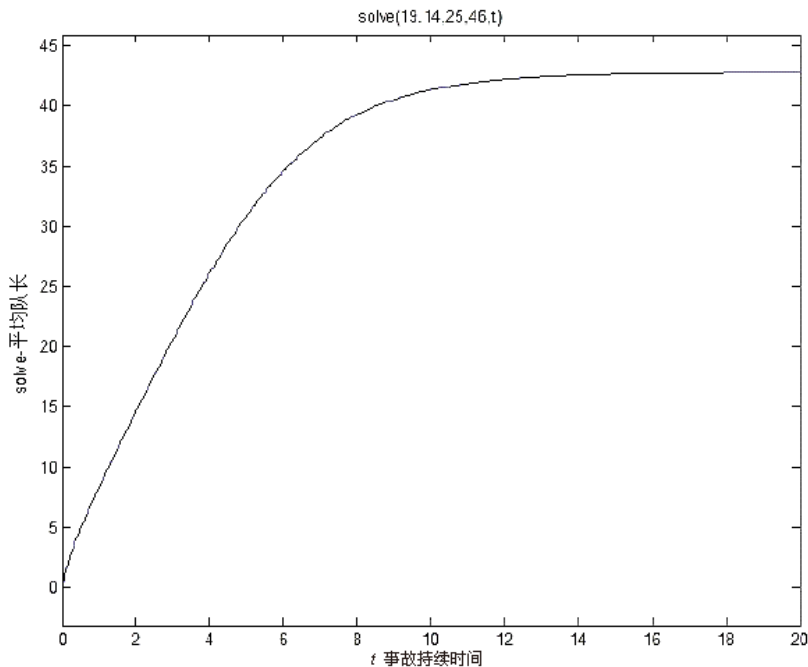


图 5-13



(2) 第二种模型：分段差分模型。

考虑红绿灯对流入车辆的影响，当红灯的时候上游路口流入量为  $\lambda_1'$ ，当绿灯的时候上游路口流入量为  $\lambda_2'$ ，小区路口流入量  $a_1$ ，小区路口流出量  $a_2$ ，道路通行能力  $\mu$ ；

考虑到红灯和绿灯的持续时间都是 30s，即  $\Delta t = 30\text{s}$ ，队长的变化量由于红绿灯不同时刻不同而不同，由于视频 1 中事故发生时正好是红灯亮的时候，于是：

$$s(t + \Delta t) - s(t) = \begin{cases} \lambda_1' - \mu + a_1 - a_2 & [t \in (60k, 60k + 30)] \\ \lambda_2' - \mu + a_1 - a_2 & [t \in (60k + 30, 60k + 60)] \end{cases}$$

其中事故持续时间  $t$  的单位是秒， $k=1,2,\dots$

$$\text{令 } \omega_1 = \lambda_1' - \mu + a_1 + a_2, \omega_2 = \lambda_2' - \mu + a_1 + a_2$$

解此差分方程得到

$$s(t) = \begin{cases} s(0) + k(\omega_1 + \omega_2) + \omega_1(t - 60k) & [t \in (60k, 60k + 30)] \\ s(0) + k(\omega_1 + \omega_2) + \omega_1 + \omega_2(t - 60k - 30) & [t \in (60k + 30, 60k + 60)] \end{cases}$$

MatLab 做出图中（程序见附件 2）得到视频 1 的队长时间图如图 5-14 所示。

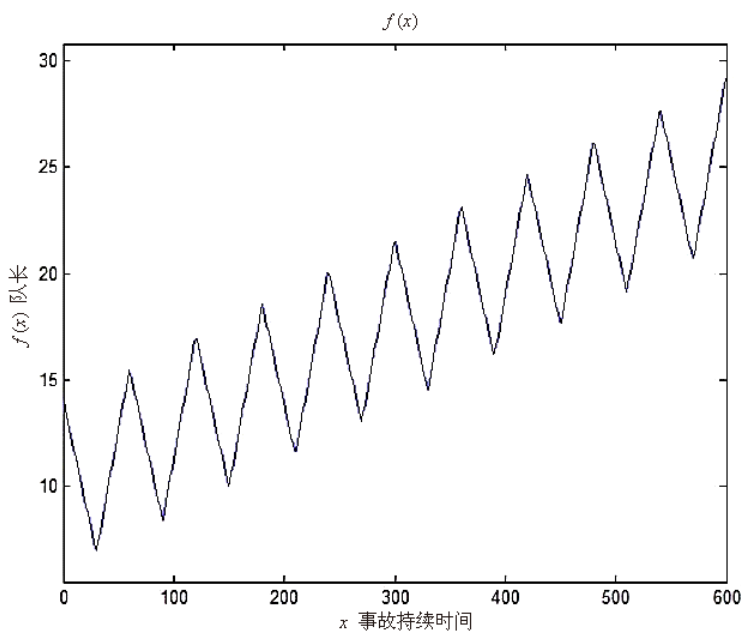


图 5-14

可以发现，当红灯的时候，加入排队的车辆少，排队车辆在减少，绿灯的时候有大量车辆涌入，排队车辆增多，总体来说，通行能力小于车辆流入速度，总体呈队长上升的趋势，队长周期性变化，细致地描绘的车队长变化过程，很好地反应了客观事实。从表达式



中还可以看出：实际通行能力越大路段车辆排队长度越小，或者说增长越慢；路段上游车流量越大路段车辆排队长度越大，或者说增长越快。

### 3. 问题 3 的解答

无论是从采用非稳态排队论模型得到的

$$m(t) = \sum_{i=0}^n iP_i(t) = \sum_{i=0}^n i \sum_{k=0}^n c_k e^{x_k t} r(k, i)$$

还是从采用分段差分方程模型得到的

$$s(t) = \begin{cases} s(0) + k(\omega_1 + \omega_2) + \omega_1(t - 60k) & [t \in (60k, 60k + 30)] \\ s(0) + k(\omega_1 + \omega_2) + \omega_1 + \omega_2(t - 60k - 30) & [t \in (60k + 30, 60k + 60)] \end{cases}$$

都可以得出一致的结论：实际通行能力越大，交通事故所影响的路段车辆排队长度越小，或者说增长越慢；路段上游车流量越大交通事故所影响的路段车辆排队长度越大，或者说增长越快；事故持续时间越长，交通事故所影响的路段车辆排队长度越大。再次说明两模型的合理性和实用性。

### （四）问题 4 的解答

第四问给出了一个具体的情况，在具体案例环境中应用问题三模型求解。

#### 1. 数据收集

由于两个小区均有车辆进出，因此我们在计算车辆流入的时候应该将两个小区车辆的进出一起考虑，我们以每 30s 统计在事故发生之后两个小区车辆的净流入，并按照问题 1 的方法将缺失的数据补全记小区 1 为靠近下游路口的小区，小区 2 为靠近上游路口的小区。结果如表 5-7 所列。

表 5-7

时间 $t$	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	6
小区 1	3	3	0	1	1	0	1	2	0	1	1	1
小区 2	0	0	-1	0	-2	-1	0	0	-1	0	-1	-1
时间 $t$	6.5	7	7.5	8	8.5	9	9.5	10	10.5	11	11.5	12
小区 1	1	2	0	2	0	2	0	0	0	1	0	2
小区 2	0	-3	-1	-1	0	0	0	-1	0	-1	0	0
时间 $t$	12.5	13	13.5	14	14.5	15	15.5	16	16.5	17	17.5	18
小区 1	0	1	1	1	0	0	0	1	0	0	0	1
小区 2	0	-4	-1	0	-1	-1	-1	-1	-1	0	0	0

由于问题三已经检验到达车辆率服从泊松分布，因此我们分别计算两个小区的净到达率分别为 0.89 辆/0.5min 和 -0.67 辆/0.5min。

## 2. 问题4 基于两种模型的求解

(1) 第一种方法：非稳态排队论模型。

为方便计算,先计算出堵到路口需要的车辆,考虑到车队中直走车辆所占比例为44%,所占比例最重,中间道路往往最长,所以  $S(t)=0.44 M l_0=140$ , 其中  $l_0=7$  (m) 为平均车距和平均车长之和, 求出  $M=46$ , 考虑到系统的最大容量  $M_{\max}=\frac{140 \times 3}{l_0}=60$ , 随着时间的变化, 当概率  $P_{46}(t_1)=\max P_{46}(t)$  的时候也就是排队最有可能排到路口的时候, 因为超过这个时间点的时候, 队长应该更长, 所以概率会下降, 在这个时间点之前, 由于队长还不够长, 到达的概率就会低于这点的概率。

于是用 MatLab 做出  $P_{46}(t)=\sum_{k=0}^{60} c_k e^{\lambda_k t} r(k, 46)$  关于时间的图像, 求出最大值时的  $t$  值就是到达时间。其中:  $\lambda=25, \mu=19.14$ 。

$$\begin{cases} x_0 = 0 \\ x_k = 2\sqrt{\lambda\mu} \cos \frac{k\pi}{n+1} - \lambda - \mu \end{cases} \quad (k > 0)$$

用 MatLab 可求得对应特征向量

$$\vec{r}_j = \begin{pmatrix} r_{(j,1)} \\ r_{(j,2)} \\ \dots \\ r_{(j,n)} \end{pmatrix}$$

$$\begin{cases} 0 = c_0 r_{(0,0)} + c_1 r_{(1,0)} + \dots + c_n r_{(n,0)} & (1) \\ \dots & \\ 1 = c_0 r_{(0,k)} + c_1 r_{(1,k)} + \dots + c_n r_{(n,k)} & (k) \\ \dots & \\ 0 = c_0 r_{(0,n)} + c_1 r_{(1,n)} + \dots + c_n r_{(n,n)} & (n) \end{cases}$$

(2) 第二种方法：分段差分方程模型。

针对红绿灯切换的情况, 乐观估计当事故发生是红灯的时候, 根据视频1的数据, 将路段上游车流量为 1500pcu/h (25pcu/min), 按视频1的红绿灯到达车流量比例分配为: 红

灯到达率  $\lambda_1' = 1500 \frac{1.61}{1.61+17.44} = 2.11$ ;

绿灯到达率  $\lambda_2' = 1500 \frac{17.44}{1.61+17.44} = 22.89$ , 则

$$s(t+\Delta t) - s(t) = \begin{cases} \lambda_1' - \mu - a_2 & [t \in (60k, 60k+30)] \\ \lambda_2' - \mu - a_2 & [t \in (60k+30, 60k+60)] \end{cases}$$





令  $\omega_1 = \lambda_1' - \mu - a_2, \omega_2 = \lambda_2' - \mu - a_2$ 。

解得：

$$s(t) = \begin{cases} s(0) + k(\omega_1 + \omega_2) + \omega_1(t - 60k) & [t \in (60k, 60k + 30)] \\ s(0) + k(\omega_1 + \omega_2) + \omega_1 + \omega_2(t - 60k - 30) & [t \in (60k + 30, 60k + 60)] \end{cases}$$

悲观估计当事故发生时，上游路口为绿灯的情况（耗时更长）：

$$s(t + \Delta t) - s(t) = \begin{cases} \lambda_2' - \mu - a_2 & [t \in (60k, 60k + 30)] \\ \lambda_1' - \mu - a_2 & [t \in (60k + 30, 60k + 60)] \end{cases}$$

令  $\omega_1 = \lambda_2' - \mu - a_2, \omega_2 = \lambda_1' - \mu - a_2$ 。

解得：

$$s(t) = \begin{cases} s(0) + k(\omega_1' + \omega_2') + \omega_1'(t - 60k) & [t \in (60k, 60k + 30)] \\ s(0) + k(\omega_1' + \omega_2') + \omega_1' + \omega_2'(t - 60k - 30) & [t \in (60k + 30, 60k + 60)] \end{cases}$$

### 3. 问题 4 的解答

对于排队论模型，用 MatLab 做出图 5-15（程序见附录 3），可知当状态为 46 辆车时的最大概率在 6.656min 到达，所以经过 6.656min 车辆排队长度将到达上游路口。

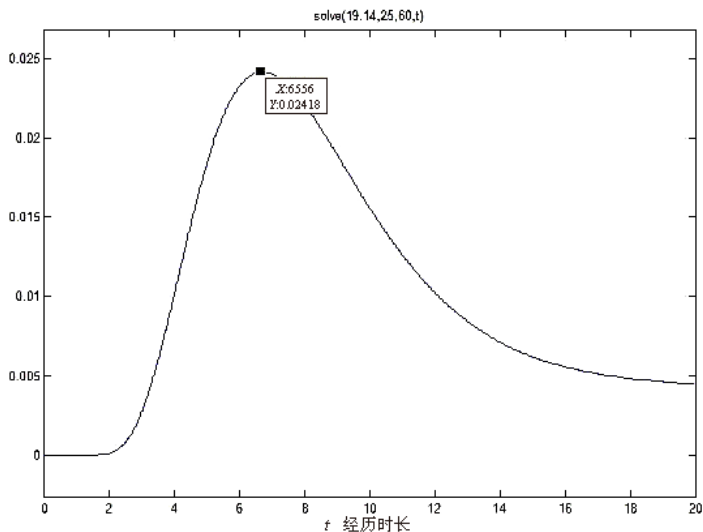


图 5-15

对于考虑红绿灯的分段差分方程模型，乐观估计当事故发生时是红灯时，用 MatLab 做出图 5-16（程序见附件 3），由图 5-16 可知，在 478.1s 的时候，第一次到达 46 辆车，所以经过 7.97min 车辆排队长度将到达上游路口。

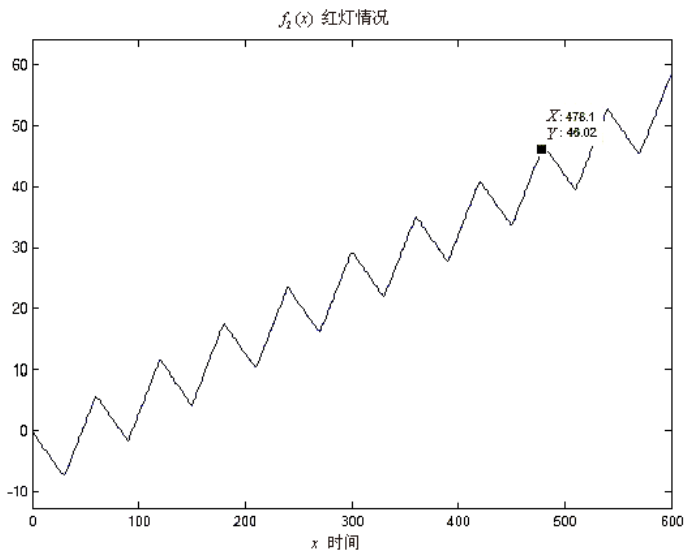


图 5-16

悲观估计当事故发生时是绿灯时，用 MatLab 做出图 5-17（程序见附录 3），由图可知，在 386.2s 的时候，第一次到达 46 辆车，所以经过 6.42min 车辆排队长度将到达上游路口。

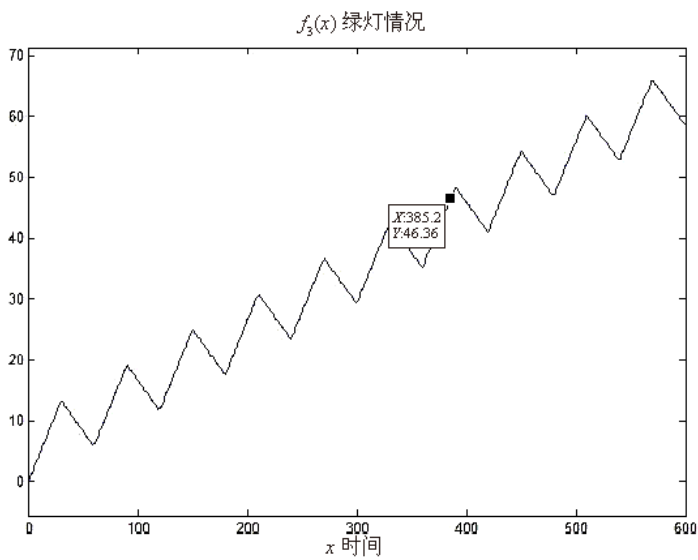


图 5-17

综上所述，根据排队论模型，从事故发生开始，经过 6.656min，车辆排队长度将到达上游路口；根据差分方程模型，事故发生时上游路口刚好是绿灯时，需要经过 6.42min，事故发生时上游路口刚好是红灯时，需要经过 7.97min。



#### 四、论文中程序的解析

论文中用到的程序主要用 Excel 和 MatLab 实现,对问题的解答也主要是用 MatLab 作图来直观反应并解释问题。

##### (一) Gibbs 抽样仿真的实现

问题一用 Excel 软件实现 Gibbs 抽样仿真,实现语句如下:

- ① 用 rand()语句得到 0—1 间随机分布的随机数。
- ② 再用如下语句实现 Gibbs 抽样仿真。

```
=IF(E2=1,IF(D3<0.0833,1,IF(D3<0.25,2,IF(D3<0.3333,3,
IF(D3<0.5,4,IF(D3<0.6667,5,IF(D3<0.9167,6,IF(D3<1,7,8)))))),
IF(D3<0.0833,1,IF(D3<0.3333,2,IF(D3<0.4167,3,IF(D3<0.6667,4,
IF(D3<0.8333,5,IF(D3<0.8333,6,IF(D3<0.9167,7,8)))))))

=IF(E2=2,IF(D3<0.0833,1,IF(D3<0.25,2,IF(D3<0.3333,3,
IF(D3<0.5,4,IF(D3<0.6667,5,IF(D3<0.9167,6,IF(D3<1,7,8)))))),
IF(D3<0.0833,1,IF(D3<0.3333,2,IF(D3<0.4167,3,IF(D3<0.6667,4,
IF(D3<0.8333,5,IF(D3<0.8333,6,IF(D3<0.9167,7,8)))))))
```

##### (二) 排队情况的 MatLab 实现

```
%调用 f 函数,输入命令 ezplot('f(t)',[0,600]);输出时间-队长曲线
%第三问视频 1 中的排队情况
function m=f(t)
y=0;
for k=0:20
    %时间上限为 20
    if t<(30+60*k)&&t>=(60*k)
        y=14+k*1.53-7.15*(t-60*k)/30;
        %14 是初始车辆数
    elseif t>=(30+60*k)&&t<(60*k+60)
        y=14+k*1.53-7.15+8.68*(t-30-60*k)/30;
    else
        y=y;
    end
end
m=y;
end
```

##### (三) 非稳态排队论和分段差分方程的 MatLab 程序

问题三、问题四建立了排队论模型和差分方程模型,MatLab 程序实现如下:

## 1. 排队论模型的 MatLab 实现, 输出为队长

```

%调用 solve 函数用 ezplot 画曲线
%3.1 首先输入命令 ezplot('solve(x,25,46,5)',[10,30]);
%3.2 其次输入命令 ezplot('solve(19.14,x,46,5)',[10,40]);
%3.3 最后输入命令 ezplot('solve(19.14,25,46,t)',[0,20]);
%得到三张图表示路段车辆排队长度与事故横断面实际通行能力、事故持续时间、路段上游车流量间的关系
function uu=solve1(u,w,n,t)
%输入参数的含义:
    %u 表示单位时间通过的车辆数 mu, 即服务完成的对象个数. 输入 19.14
    %w 表示单位时间到达车辆数 lambda. 输入 25
    %n 表示排队长度. 输入 46
    %t 表示排队时间
a0=1;
%赋初值
%通过大循环求解 a(n-1, ,n)
for f=1:n
    e(f)=sqrt(4*w*u)*cos(f*pi/(n+1));
    a(1,f)=e(f)/(e(f)-u);
    for i=2:n-1
        a(i,f)=(w-e(f)-(w/a(i-1,f)))/(w+u-e(f)-(w/a(i-1,f)));
    end
    for i=2:n
        s=a0;
        p=1;
        for k=1:i-1
            s=a(k,f)*s;
        end
        if i<n
            for o=1:i
                p=(a(o,f)-1)*p;
            end
        else
            for o=1:i-1
                p=(a(o,f)-1)*p;
            end
        end
        rr(i,f)=s/p;
    end
    rr(1,f)=a0/(a(1,f)-1);
end rr(n,:)= -rr(n,:);

%求解 x, 已知矩阵 Q 的特征值
%构造矩阵 Q
Q=zeros(n+1,n+1); Q(1,1)=-w; Q(1,2)=u; Q(n+1,n)=w; Q(n+1,n+1)=-u;
for i=1:n-1
    Q(i+1,i)=w;
    Q(i+1,i+1)=-w-u;

```





```

        Q(i+1,i+2)=u;
    end

    %用 MatLab 工具箱函数 eig 求特征值 x,特征向量 r
    x=eig(Q); [r,aa]=eig(Q);
    %构造矩阵 B
    B(1)=1; for i=1:n
        B(i+1)=0;
    end B=B';
    %解线性方程 r*c=B
    c=r\B;
    c=c';
    %su=c(1)*(w/u)^n;
    %循环求平均队长达到 n 值的概率
    ss=0;
    for k=1:n+1
        ss=c(k)*2.718^(x(k)*t)*r(46,k)+ss;
    end
    m=ss;
    %得到结果, 平均队长达到 n 值的概率为 m
    %求平均队长
    uu=0;
    for i=1:n+1
        ss=0;
        for k=1:n+1
            ss=c(k)*2.718^(x(k)*t)*r(i,k)+ss;
        end
        uu=(i-1)*ss+uu;
    end
end
end

```

## 2. 排队论模型的 MatLab 实现, 输出为概率

```

%调用 solve1 函数用 ezplot 画曲线, 输入命令 ezplot('solve2(19.14,25,46,t)',
[0,20]);
%得到车辆排队长度将到达上游路口的时间, 即图像最高点
function uu=solve2(u,w,n,t)
%输入参数的含义:
    %u 表示单位时间通过的车辆数 mu, 即服务完成的对象个数. 输入 19.14
    %w 表示单位时间到达车辆数 lambda. 输入 25
    %n 表示排队长度. 输入 46
    %t 表示排队时间
a0=1;
%赋初值
%通过大循环求解 a(n-1, ,n)
for f=1:n
    e(f)=sqrt(4*w*u)*cos(f*pi/(n+1));
    a(1,f)=e(f)/(e(f)-u);

```



```

for i=2:n-1
    a(i,f)=(w-e(f)-(w/a(i-1,f)))/(w+u-e(f)-(w/a(i-1,f)));
end
for i=2:n
    s=a0;
    p=1;
    for k=1:i-1
        s=a(k,f)*s;
    end
    if i<n
        for o=1:i
            p=(a(o,f)-1)*p;
        end
    else
        for o=1:i-1
            p=(a(o,f)-1)*p;
        end
    end
    rr(i,f)=s/p;
end
rr(1,f)=a0/(a(1,f)-1);
end rr(n,:)=-rr(n,:);

%求解 x, 已知矩阵 Q 的特征值
%构造矩阵 Q
Q=zeros(n+1,n+1); Q(1,1)=-w; Q(1,2)=u; Q(n+1,n)=w; Q(n+1,n+1)=-u;
for i=1:n-1
    Q(i+1,i)=w;
    Q(i+1,i+1)=-w-u;
    Q(i+1,i+2)=u;
end

%用 MatLab 工具箱函数 eig 求特征值 x, 特征向量 r
x=eig(Q); [r,aa]=eig(Q);
%构造矩阵 B
B(1)=1; for i=1:n
    B(i+1)=0;
end B=B';
%解线性方程 r*c=B
c=r\B; c=c';
%su=c(1)*(w/u)^n;
%循环求平均队长达到 n 值的概率
ss=0;
for k=1:n+1
    ss=c(k)*2.718^(x(k)*t)*r(46,k)+ss;
end
m=ss;
%得到结果, 平均队长达到 n 值的概率为 m

```



### 3. 分段差分方程 MatLab 实现（绿灯）

```
%调用 f3 函数，输入命令 ezplot('f3(t)',[0,600]);输出时间-队长曲线
%第四问差分模型，求解当发生事故时上游路口恰好绿灯的情况下，车辆数目与时间的关系
%做出图像，通过车辆数目实际值的分析，得到车辆排队长度将到达上游路口的时间
function m=f3(t) y=0; for k=0:90
    %时间上限为 90
    if t<(30+60*k)&&t>=(60*k)
        y=k*5.86+13.32*(t-60*k)/30;
    elseif t>=(30+60*k)&&t<(60*k+60)
        y=k*5.86+13.32-7.46*(t-30-60*k)/30;
        %13.32-7.46=5.82
        %其中 22.86 为绿灯时候每半分钟的到达的车辆，2.11 为红灯时候每半分钟到达的车辆
        %半分钟的车辆通行能力为 8.9 辆，从小区离开的车辆每半分钟为 0.67
        %2.11-8.9-0.67=-7.46，22.89-8.9-0.67=13.32
    else
        y=y;
    end
end
m=y;
end
```

### 4. 分段差分方程 MatLab 实现（红灯）

```
%调用 f2 函数，输入命令 ezplot('f2(t)',[0,600]);输出时间-队长曲线
%第四问差分模型，求解当发生事故时上游路口恰好红灯的情况下，车辆数目与时间的关系
%做出图像，通过车辆数目实际值的分析，得到车辆排队长度将到达上游路口的时间
function m=f2(t) y=0; for k=0:90
    %时间上限为 90
    if t<(30+60*k)&&t>=(60*k)
        y=k*5.86-7.46*(t-60*k)/30;
    else
        if t>=(30+60*k)&&t<(60*k+60)
            y=k*5.86-7.46+13.32*(t-30-60*k)/30;
            %13.32-7.46=5.82
            %其中 22.86 为绿灯时候每半分钟的到达的车辆，2.11 为红灯时候每半分钟到达的车辆
            %半分钟的车辆通行能力为 8.9 辆，从小区离开的车辆每半分钟为 0.67
            %2.11-8.9-0.67=-7.46，22.89-8.9-0.67=13.32
        else
            y=y;
        end
    end
end
m=y;
end
```

### 5.1.3 论文点评

#### 一、整体点评

本篇论文在问题分析方面表现突出,能够抓住问题本质和关键点,建立合适的模型,对题目提出的问题也给出了好的解答。具体到每一问上来说,论文概括分析如下:

针对问题1,论文中通过对视频1中交通事故发生至撤离期间,各种数据的采集,确定了事故横断面实际通行能力,运用Gibbs抽样仿真方法,通过Excel软件解决数据缺失问题,并用eviews进行ARMA模型拟合,发现实际通行能力的变化过程为 $y_t = -2.0232 + \epsilon_t - 0.7243\epsilon_{t-1} - 0.2757\epsilon_{t-2}$ 。论文选用了合适的数据处理方法,并用简答的编程语言实现,再通过建模作图直观地回答了第一问的问题。

针对问题2,论文结合视频1,采用与问题1同样的方法对视频2进行数据采集和分析,得到了横断面的实际通行能力,并将视频1和视频2仔细对比,通过视频1和视频2中对交通事故发生至撤离期间,事故所处横断面实际通行能力,运用SPSS软件进行两独立样本的曼-惠特尼U检验,再根据路段附近交通设置,车辆流向比例,司机心理,周围地形等因素,分析出产生差异的原因主要是在不同车道车辆流量的比例。

针对问题3,通过分析视频1塞车情况,分别建立非稳态排队论模型和分段差分方程模型,并运用MatLab软件编程绘制图像,解释出视频1中路段车辆排队长度与事故横断面实际通行能力、事故持续时间、路段上游车流量间的关系为:实际通行能力越大交通事故所影响的路段车辆排队长度越小,或者说路段车辆排队长度增长越慢;路段上游车流量越大交通事故所影响的路段车辆排队长度越大,或者说路段车辆排队长度增长越快;事故持续时间越长,交通事故所影响的路段车辆排队长度越大。

针对问题四,利用问题3所建立的模型和题目所给数据,如果采用非稳态排队论模型,运用MatLab软件求解,得出从事故发生开始,经过6.656min,车辆排队长度将到达上游路口;如果通过分段差分方程模型,事故发生时上游路口刚好是绿灯时需要经过6.42min,事故发生时上游路口刚好是红灯时需要经过7.97min,与上一模型所得结果相吻合。说明两模型有很高的合理性和实用性。

#### 二、重点点评

对于论文中提出的主要问题,从不同的角度建立了两个模型来解释视频1中的车辆排队长度与其他三种因素之间的关系。首先,排队模型根据灭生原理合理的解释了因交通事故造成的堵塞问题,并且利用非稳态的解法求解出了排队模型的通解。而且这种排队模型能够较好的考虑到当队长到达小区1路口时候,由于小区1的车辆无法进出而产生的另一种排队模型。另一种分段差分方程的模型则比较好的考虑了红绿灯周期的问题,根据红绿灯交替的周期建立了模型,并且在问题四求解的时候可以极端的考虑最好和最差情况下所需要的时间,使得模型对问题的分析更全面,更科学,更严谨。





## 5.2 2011 CUMCM-B

### 5.2.1 问题综述

“有困难找警察”，是家喻户晓的一句流行语。警察肩负着刑事执法、治安管理、交通管理、服务群众四大职能。为了更有效地贯彻实施这些职能，需要在市区的一些交通要道和重要部位设置交巡警服务平台。每个交巡警服务平台的职能和警力配备基本相同。由于警务资源是有限的，如何根据城市的实际情况与需求合理地设置交巡警服务平台、分配各平台的管辖范围、调度警务资源是警务部门面临的一个实际课题。

试就某市设置交巡警服务平台的相关情况，建立数学模型分析研究下面的问题：

(1) 附件 1（见原题）中给出了该市中心城区 A 的交通网络和现有的 20 个交巡警服务平台的设置情况示意图，相关的数据信息见附件 2。请为各交巡警服务平台分配管辖范围，使其在所管辖的范围内出现突发事件时，尽量能在 3min 内有交巡警（警车的时速为 60km/h）到达事发地。

对于重大突发事件，需要调度全区 20 个交巡警服务平台的警力资源，对进出该区的 13 条交通要道实现快速全封锁。实际中一个平台的警力最多封锁一个路口，请给出该区交巡警服务平台警力合理的调度方案。

根据现有交巡警服务平台的工作量不均衡和有些地方出警时间过长的实际情况，拟在该区内再增加 2 至 5 个平台，请确定需要增加平台的具体个数和位置。

(2) 针对全市（主城六区 A, B, C, D, E, F）的具体情况，按照设置交巡警服务平台的原则和任务，分析研究该市现有交巡警服务平台设置方案的合理性。如果有明显不合理，请给出解决方案。

如果该市地点 P（第 32 个节点）处发生了重大刑事案件，在案发 3min 后接到报警，犯罪嫌疑人已驾车逃跑。为了快速搜捕嫌疑犯，请给出调度全市交巡警服务平台警力资源的最佳围堵方案。

### 5.2.2 分析与建模及求解

#### 问题 1

对该问题的解决，我们先建立数学模型，将需要达到的目标，包括到达事发地的时间尽量短，各服务平台的工作量尽量均衡，用目标函数表达出来，同时将需要满足的约束也表达出来，构成合适的数学模型。然后讨论求解算法，最后给出具体的计算结果。

我们的路口管辖分配方案分为两步。

第一步：对  $T_j > 3$  的路口，分配给到达第  $j$  个路口时间最少的平台。

第二步：对  $T_j \leq 3$  的所有路口，根据尽量使各平台分配的任务量尽量均衡的原则进行



优化。

计算得到 3 分钟不能到达的路口及所属平台信息见表 5-8 所示。

表 5-8

序 号	路 口	任 务 量	所属平台	最短时间（分钟）
1	28	1.3	15	4.75
2	29	1.4	15	5.70
3	38	1.2	16	3.41
4	39	1.4	2	3.68
5	61	0.6	7	4.19
6	92	0.8	20	3.60

因此，我们将这 6 个路口分配给最近的平台，剩下的 86 个路口建立模型，根据任务量尽量均衡的原则进行优化。

实现的 MatLab 程序见附录 1 的 b2011\_1.m，利用 Floyd 算法计算 92 个路口的最短路矩阵，输出表 1 数据。输出 20 个平台和剩余 86 个待分配路口的最短路矩阵文件 dt1.txt，86 个路口的任务量文件 ft1.txt。以及 20 个平台和 13 个交通要道的距离矩阵文件 dis1.txt，该数据文件用于第二问计算。

1. 模型建立

设有 20 个平台，路口有 86 个。

$d_{ij}$  表示第  $i$  个平台与第  $j$  个路口之间的最短路，由 Floyd 算法求得。

$$i = 1, 2, \dots, 20; j = 1, 2, \dots, 86$$

$v$  为警车行驶速度。这里取  $v = 60 \text{ km/h} = 1000 \text{ m/min}$ 。

$$\text{建立决策变量 } x_{ij} = \begin{cases} 1 & \text{第 } j \text{ 个路口分配给第 } i \text{ 个平台} \\ 0 & \text{第 } j \text{ 个路口不分配给第 } i \text{ 个平台} \end{cases}$$

每个路口只分配给一个平台，有约束：

$$\sum_{i=1}^{20} x_{ij} = 1 \quad j = 1, 2, \dots, 86$$

每个平台至少管理一个路口，有约束：

$$\sum_{j=1}^{86} x_{ij} \geq 1 \quad i = 1, 2, \dots, 20$$

每个平台管理自己的路口，则有：

$$x_{ii} = 1, \quad i = 1, 2, \dots, 20$$

第  $j$  个路口到指派的平台的时间为  $t_j$ ，满足：

$$t_j = \sum_{i=1}^{20} x_{ij} \cdot d_{ij} / v \quad j = 1, 2, \dots, 86$$



其时间应满足不超过 3min, 则有:

$$t_j \leq 3 \quad j=1, 2, \dots, 86$$

计算第  $i$  个平台分配的路口数的任务量。

设已知第  $j$  个路口的任务量 (平均每天的发生报警案件数量) 为  $f_j$ ,  $j=1, 2, \dots, 86$ , 则第  $i$  个平台已经分配到的任务量:

$$s_i = w_i + \sum_{j=1}^{86} f_j \cdot x_{ij} \quad i=1, 2, \dots, 20$$

$w_i$  为各平台已分路口数的任务量。由表 5-8 知为  $w_2 = 1.4$ ,  $w_7 = 0.6$ ,  $w_{15} = 1.4 + 1.3 = 2.7$ ,  $w_{16} = 1.2$ ,  $w_{20} = 0.8$ , 其余为 0。

各平台分配的任务量平均值为:

$$\bar{s} = \frac{\sum_{i=1}^{20} s_i}{20}$$

我们的目标是各平台分配的任务量的标准差尽量小。即:

$$\min Z = \sqrt{\frac{\sum_{i=1}^{20} (s_i - \bar{s})^2}{20 - 1}}$$

因此我们得到的综合模型为:

$$\begin{aligned} \min Z = & \sqrt{\frac{\sum_{i=1}^{20} (s_i - \bar{s})^2}{20 - 1}} \\ \text{s.t.} \quad & \begin{cases} \sum_{i=1}^{20} x_{ij} = 1 & j=1, 2, \dots, 86 \\ \sum_{j=1}^{86} x_{ij} \geq 1 & i=1, 2, \dots, 20 \\ x_{ii} = 1 & i=1, 2, \dots, 20 \\ t_j = \sum_{i=1}^{20} x_{ij} \cdot d_{ij} / v & j=1, 2, \dots, 86 \\ t_j \leq 3 & j=1, 2, \dots, 86 \\ s_i = w_i + \sum_{j=1}^{86} f_j \cdot x_{ij} & i=1, 2, \dots, 20 \\ \bar{s} = \frac{\sum_{i=1}^{20} s_i}{20} \\ x_{ij} = 0 \text{ 或 } 1 & i=1, 2, \dots, 20; j=1, 2, \dots, 86 \end{cases} \end{aligned}$$

Lingo 程序见附录 2 的 b2011\_1.lg4。  
采用 Lingo12 优化很快得到  $Z=1.76$ ，表示每个平台的任务量平均有 1.76 件的波动。  
各平台管辖的路口的分配方案见表 5-9 所示。

表 5-9

平 台	管辖的路口	总任务量	最长时间 (min)
1	1,64,71,72,77,78,79,80	7.60	2.60
2	2,40,69,74	7.40	2.53
3	3,44,54,55,70,76	7.20	2.97
4	4,57,60,62,63,65,66	7.30	2.84
5	5,49,53,56,59	6.10	2.08
6	6,50,51,52,58	6.10	2.38
7	7,30,48	6.50	1.29
8	8,32,33,47	6.90	2.08
9	9,35,36,37,45	6.10	1.43
10	10,	1.60	0
11	11,26,27	4.60	1.64
12	12,25	4.00	1.79
13	13,21,22,23,24	8.50	2.71
14	14	2.50	0
15	15,31	6.40	2.97
16	16,34,46	6.70	2.38
17	17,41,42,43	7.00	1.79
18	18,85,87,89,90,91	7.40	2.63
19	19,67,68,73,75,82,83	7.20	2.79
20	20,81,84,86,88	7.40	2.68

问题 2

给出了该市中心城区 A 的交通网络和现有的 20 个交巡警服务平台的设置情况示意图，  
相关的数据信息见附件 2（见原题）。对于重大突发事件，需要调度全区 20 个交巡警服  
务平台的警力资源，对进出该区的 13 条交通要道实现快速全封锁。实际中一个平台的警力最  
多封锁一个路口，我们的目的是给出该区交巡警服务平台警力合理的调度方案。

对该问题，我们先建立最优的调度模型，使各服务平台到达交通要道的时间尽可能的短。  
然后讨论求解算法，最后给出具体的计算结果。

1. 模型建立

对于重大突发事件，需要调度城市的交巡警服务平台的警力资源，对进出该区的多条  
交通要道实现快速全封锁。采用的模型如下。

设该市有 20 个平台，要封锁的交通要道有 13 个。  
 $d_{ij}$  表示第  $i$  个平台与第  $j$  个交通要道之间的最短路，由 Floyd 算法求得。

$v$  为警车行驶速度。这里取  $v = 60 \text{ km/h} = 1000 \text{ m/min}$

设 0~1 决策变量  $x_{ij} = \begin{cases} 1 & \text{第 } j \text{ 个路口分配给第 } i \text{ 个平台围堵} \\ 0 & \text{第 } j \text{ 个路口不分配给第 } i \text{ 个平台围堵} \end{cases}$



对每个交通要道，需要且只需要分配一个平台的警力，则有：

$$\sum_{i=1}^{20} x_{ij} = 1 \quad j = 1, 2, \dots, 13$$

每个交巡警服务平台的警力最多到一个交通要道去围堵，因此有：

$$\sum_{j=1}^{13} x_{ij} \leq 1 \quad i = 1, 2, \dots, 20$$

设  $T_j$  表示到第  $j$  个路口的时间。则有：

$$T_j = \sum_{i=1}^{20} d_{ij} x_{ij} / v \quad j = 1, 2, \dots, 13$$

我们选取的第一目标是到交通要道的最长时间最小化，这样可使最长时间尽量小。则第一目标函数为：

$$\min Z_1 = \max_{1 \leq j \leq 13} T_j$$

当最长时间最小情况下，我们同时对小于最长时间的分配方式进行优化，使到达各交通要道的平均时间最小。则第二目标函数为：

$$\min Z_2 = \frac{\sum_{j=1}^{13} T_j}{13}$$

综合上述，我们建立的的综合模型为：

$$\begin{aligned} \min Z_1 &= \max_{1 \leq j \leq 13} T_j \\ \min Z_2 &= \frac{\sum_{j=1}^{13} T_j}{13} \\ \text{s.t.} \quad &\begin{cases} \sum_{i=1}^{20} x_{ij} = 1 & j = 1, 2, \dots, 13 \\ \sum_{j=1}^{13} x_{ij} \leq 1 & i = 1, 2, \dots, 20 \\ T_j = \sum_{i=1}^{20} d_{ij} x_{ij} / v \\ x_{ij} = 0 \text{ 或 } 1 \end{cases} \end{aligned}$$

Lingo 程序见附录 3 的 b2011\_2.lg4。可先优化第一目标，得到最短时间为 8.02min。然后将  $z_1 \leq 8.02$  作为约束优化第二目标，得到最小的平均时间为 3.55min。围堵求解的最优分配方案见表 5-10 所示。

该模型由于是非线性的，利用 Lingo 求解较为费时，通常也不能保证为最优解。为此。我们也可以另外设计算法，便于快速求解，同时便于在后面的围堵问题中快速求解。



表 5-10

序 号	交通要道号	分配的平台号	到达时间 (min)
1	12	12	0
2	14	16	6.74
3	16	9	1.53
4	21	14	3.26
5	22	10	7.71
6	23	13	0.5
7	24	11	3.81
8	28	15	4.75
9	29	7	8.02
10	30	8	3.06
11	38	2	3.98
12	48	5	2.48
13	62	4	0.35

2. 算法设计

我们的求解算法采用三步完成。第一步，先利用贪婪法尽量使各平台到达交通要道的时间尽量短。第二步，对到各交通要道的时间再进行调整，进一步优化，直到最长时间不能再减少为止。第三步，在保证最长时间不增大的情况下，对到各交通要道的平均时间进行调整，直到不再减小为止。

算法步骤：

- (1) 先将 13 个交通要道依次分配给最近的路口。
- (2) 将时间最长的要道与其余的要道分配的平台对换，若最长时间可以减少，则对换。实际中可在对换后，另一要道在剩余平台中选择最近的平台。
- (3) 重复 2，直到最长时间不再减少为止。

对平均时间的减少采用同样的方法。直到总时间或平均时间不再减小结束。输出各路口对应平台及时间，平均时间。

采用步骤 (1) 中贪婪法求得最长时间为 13.67min，平均时间为 3.95 min。采用步骤 (2) 中减少最长时间的算法，经过 5 次调整，最长时间达到最小。最长时间为 8.02 min，平均时间为 3.78 min。计算的结果见表 5-11 所示。遗憾的是没有达到最小的 3.55 min。但该方法的优点是计算速度快。不到 1s 就可以计算出结果，比 Lingo 计算的快多了。

表 5-11

序 号	交通要道号	分配的平台号	到达时间 (min)
1	12	10	7.59
2	14	16	6.74
3	16	9	1.53
4	21	14	3.26
5	22	11	3.27
6	23	13	0.5
7	24	12	3.59
8	28	15	4.75



(续表)

序 号	交通要道号	分配的平台号	到达时间 (min)
9	29	7	8.02
10	30	8	3.06
11	38	2	3.98
12	48	5	2.48
13	62	4	0.35

对全市所有路口的围堵, 我们可建立同样模型, 分别采用 Lingo 和我们提出的贪婪算法计算。

### 3. 全市路口的围堵计算

全市有交巡警平台 80 个, 全市出入口位置有 17 个。要封锁的交通要道  $m=17$  个, 其集合为  $J=\{151, 153, 177, 202, 203, 264, 317, 325, 328, 332, 362, 387, 418, 483, 541, 572, 578\}$ 。可供分配的平台  $n=80$  个, 其集合为  $P=\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 93, 94, 95, 96, 97, 98, 99, 100, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 320, 321, 322, 323, 324, 325, 326, 327, 328, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485\}$ 。

采用步骤 (1) 中贪婪法求得最长时间为 12.68 min, 平均时间为 5.07 min。该结果通过调整并不能减少最长时间, 可以验证, 该分配方案中, 除 202 号交通要道外, 其余各交通要道都分配的到达时间最短的平台。而到达 202 号交通要道的时间最少的平台为 177, 次之为 175, 177 号平台分配 177 号交通要道最佳, 从而分配 175 号平台 202 号交通要道最佳。总的计算结果是最长时间最小为 12.68 min, 平均时间最小为 5.07 min。可验证该结果即为最优结果。详细结果见表 5-12 所示。

表 5-12

序 号	交通要道号	分配的平台号	到达时间 (min)
1	151	96	3.19
2	153	99	4.47
3	177	177	0
4	202	175	11.62
5	203	178	4.45
6	264	166	6.62
7	317	181	5.48
8	325	325	0
9	328	328	0
10	332	386	7.62
11	362	323	8.11
12	387	100	12.68
13	418	379	7.419
14	483	483	0
15	541	484	7.04
16	572	485	1.66
17	578	479	5.76



对比：利用 Lingo12 求解，耗时 41 min，得到局部最优解为 35.6 min。

因此，当规模变大时，采用简单的贪婪算法有时候更方便快速。

### 问题 3

给出了该市中心城区 A 的交通网络和现有的 20 个交巡警服务平台的设置情况示意图，相关的数据信息见附件 2。根据现有交巡警服务平台的工作量不均衡和有些地方出警时间过长的实际情况，拟在该区内再增加 2 至 5 个平台，请确定需要增加平台的具体个数和位置。

对该问题，我们先建立最优的调度模型，使各服务平台到达交通要道的时间尽量短，各平台的任务量尽量均衡。然后讨论求解算法，并最后给出具体的计算结果。

#### 1. 模型建立

根据现有交巡警服务平台的工作量不均衡和有些地方出警时间过长的实际情况，需要在一个区内新增加一些平台。我们需要确定新增加平台的具体个数和位置。

设该市要巡视的路口有 92 个，其序号为  $j=1,2,\dots,92$ 。交巡警服务平台现在已有 20 个，其序号为  $i=1,2,\dots,20$ 。

$d_{ij}$  表示第  $i$  个平台与第  $j$  个路口之间的最短路，由 Floyd 算法求得。

$v$  为警车行驶速度，这里取  $v=60\text{ km/h}=1000\text{m/min}$ 。

设增加平台数为  $k$  个，则总共有平台  $20+k$  个。且所有平台都在 92 个路口中选取。每个路口的日平均案件量为  $f_j, j=1,2,\dots,92$ 。

下面我们建立新增平台的多目标规划模型。

设 0~1 变量  $Y_j$  表示哪些路口选作平台，设

$$Y_j = \begin{cases} 1 & \text{第 } j \text{ 个路口选作平台} \\ 0 & \text{第 } j \text{ 个路口不选作平台} \end{cases} \quad j=1,2,\dots,92$$

建立 0~1 决策变量  $x_{ij}$  表示平台处理路口的情况。设

$$x_{ij} = \begin{cases} 1 & \text{第 } i \text{ 平台处理第 } j \text{ 个路口} \\ 0 & \text{第 } i \text{ 平台不处理第 } j \text{ 个路口} \end{cases} \quad i,j=1,2,\dots,92$$

第  $j$  路口需要而且只需要一个平台处理。则有：

$$\sum_{i=1}^{92} x_{ij} = 1 \quad j=1,2,\dots,92$$

平台数总共为  $n+k$  个，则有：

$$\sum_{j=1}^{92} y_j = 20+k$$

其中已经有 20 个平台，且为前 20 个，因此  $y_j=1, j=1,2,\dots,20$ 。

只有当第  $i$  个路口选作平台才能去处理第  $j$  路口，因此有：





$$x_{ij} \leq y_i \quad i, j = 1, 2, \dots, 92$$

考虑各平台警力到各路口的时间尽量小。我们计算到达各路口的时间。到达路口  $j$  时间为:

$$T_j = \sum_{i=1}^{92} x_{ij} \cdot d_{ij} / v \quad j = 1, 2, \dots, 92$$

我们的目标是让最长时间最小, 因此有:

$$\min Z_1 = \max_{1 \leq j \leq 92} T_j$$

对任务量, 我们考虑将该点的发案数作为其度量值, 则第  $i$  个平台处理的业务量为:

$$w_i = \sum_{j=1}^{92} x_{ij} \cdot f_j \quad i = 1, 2, \dots, 92$$

当  $y_i = 0$  时, 所有  $x_{ij} = 0, j = 1, 2, \dots, 92$ 。则  $w_i = 0$ 。即该点没有选作平台, 其业务量为 0。  
平均业务量为:

$$\bar{w} = \frac{\sum_{i=1}^{92} w_i}{20 + k}$$

下面计算各平台业务量的离差平方和。在离差平方和中, 其计算式为

$$S = \sum_{i=1}^{92} (w_i - \bar{w})^2$$

但这里我们是对业务量  $w_i > 0$  的  $20 + k$  个平台计算其离差平方和, 因此其计算式为:

$$S = \sum_{i=1}^{92} w_i^2 - (20 + k) \cdot \bar{w}^2$$

样本标准差为:

$$\sigma = \sqrt{S / (20 + k - 1)}$$

我们的目标是尽量使各平台的业务量均衡, 因此有:

$$\min Z_2 = \min \sqrt{S / (n + k - 1)}$$

综合上述。我们得到的总模型为:

$$\min Z_1 = \max_{1 \leq j \leq 92} T_j$$

$$\min Z_2 = \min \sqrt{S / (n + k - 1)}$$

$$\begin{cases}
 \sum_{i=1}^{92} x_{ij} = 1 & j = 1, 2, \dots, 92 \\
 \sum_{i=1}^{92} y_i = 20 + k \\
 w_i = \sum_{j=1}^{92} x_{ij} \cdot f_j & i = 1, 2, \dots, 92 \\
 \bar{w} = \frac{\sum_{i=1}^{92} w_i}{20 + k} \\
 S = \sum_{i=1}^{92} w_i^2 - (20 + k) \cdot \bar{w}^2 \\
 T_j = \sum_{i=1}^{92} x_{ij} \cdot d_{ij} / v & j = 1, 2, \dots, 92 \\
 x_{ij} \leq y_i & i, j = 1, 2, \dots, 92 \\
 y_i = 1, & i = 1, 2, \dots, 20 \\
 x_{ij} = 0 \text{ 或 } 1 & i, j = 1, 2, \dots, 92 \\
 y_i = 0 \text{ 或 } 1 & i = 21, \dots, 92
 \end{cases}$$

该结果利用 Lingo 计算十分困难，考虑设计简单算法实现。

## 2. 算法设计贪婪法

每次从超过 3min 的路口中选取使最长时间最小的路口作为平台。下次再如此进行。直到每个路口到平台的时间都不超过 3min 为止。

计算程序为附件 4 中的 b2011\_3.m。

对 A 区，路口数 92，平台数 20，超过 3min 的路口 6 个。计算结果为：

Kp= 1 Plat=28 maxT=4.1902

Kp= 2 Plat=61 maxT=3.6822

Kp= 3 Plat=38 maxT=3.6013

Kp= 4 Plat=92 maxT=2.7083

增加 4 个平台最佳。

### 问题 4

考虑全市 6 个区新增平台情形，对 B 区，路口数 73，平台数 8，超过 3min 的路口数 6 超过 3min 的路口，平台，时间数值如下：

122    94    3.29

123    94    3.37

124    96    3.21

151    96    3.19





152 99 3.37

153 99 4.47

增加平台数，最长时间为

$K_p=1$  Plat=151  $\max T=3.3653$

$K_p=2$  Plat=122  $\max T=2.9863$

对 B 区，增加两个平台，最长出警时间为 2.9863min。

其他区也可采用该程序计算，即是要让最大出警时间不超过 3min，需要增加平台数较多。

全区同时考虑，全区，路口数 582，平台数 80，超过 3min 的路口数 138，则增加平台数，最长时间如下。

$K_p=1$  Plat=389  $\max T=9.1609$

$K_p=2$  Plat=329  $\max T=8.4671$

$K_p=3$  Plat=387  $\max T=8.1188$

$K_p=4$  Plat=417  $\max T=8.1069$

$K_p=5$  Plat=362  $\max T=7.8085$

$K_p=6$  Plat=248  $\max T=7.0418$

$K_p=7$  Plat=540  $\max T=6.8605$

$K_p=8$  Plat=199  $\max T=6.6223$

$K_p=9$  Plat=259  $\max T=6.6068$

$K_p=10$  Plat=505  $\max T=6.5832$

$K_p=11$  Plat=569  $\max T=6.5686$

$K_p=12$  Plat=28  $\max T=6.4259$

$K_p=13$  Plat=369  $\max T=6.4152$

$K_p=14$  Plat=261  $\max T=6.1636$

$K_p=15$  Plat=574  $\max T=6.1112$

$K_p=16$  Plat=200  $\max T=5.8259$

$K_p=17$  Plat=29  $\max T=5.7575$

$K_p=18$  Plat=578  $\max T=5.6953$

$K_p=19$  Plat=239  $\max T=5.4752$

$K_p=20$  Plat=300  $\max T=5.1506$

$K_p=21$  Plat=418  $\max T=5.0529$

$K_p=22$  Plat=370  $\max T=4.9555$

增加平台时间及对应最大出警时间图如图 5-18 所示。

从该结果来看，要让最大出警时间少于 3min，需要增加很多平台。



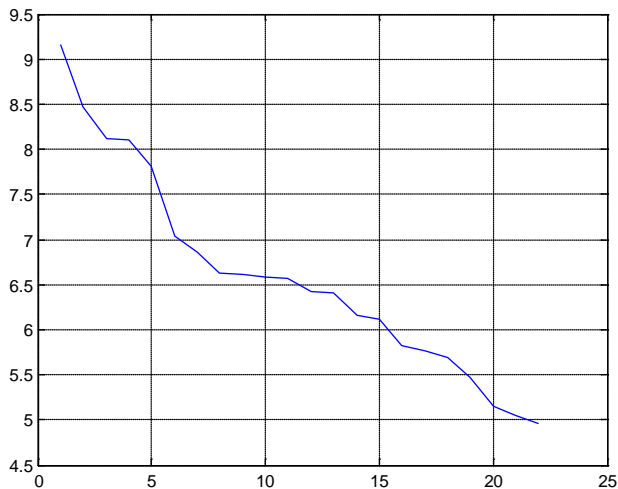


图 5-18

### 问题 5

#### 1. 给定时间 $t$ 可以包围罪犯的最小包围圈的确定

给定时间  $t$ ，给定罪犯逃跑的速度  $v$ ，计算出罪犯能逃跑的范围。这里我们采用点来表示其逃跑范围。计算其全市总路口  $N = 581$  个，罪犯在  $t \text{ min}$  内能到达的点集合，设为  $A(t)$ 。计算公式为：

$$A(t) = \{i \mid \frac{D(i, 32)}{v} \leq t, i = 12, \dots, 581\}$$

其中 32 点为案发现场，罪犯的逃跑开始点。 $D(i, 32)$  为各点到 32 点的最短距离。同时计算与  $A(t)$  相邻的点集合，设为  $B(t)$ 。

$$B(t) = \{j \mid P(i, j) = 1, i \in A(t)\}$$

其中  $P(i, j) = 1$  表示  $i$  与  $j$  直接相连； $P(i, j) = 0$  表示  $i$  与  $j$  不直接相连。该数据由附件 2 中全市交通路口的路线表单对应数据得到。

则可以包围罪犯逃跑区域  $A(t)$  的最小闭包（用路口节点表示）为  $C(t)$ ：

$$C(t) = B(t) - A(t)$$

这里的减为集合之间的减法。即从点集合  $B(t)$  中去掉  $A(t)$  中已有的点。 $C(t)$  就是给定时间可以包围罪犯的最小包围圈。

#### 2. 警察在给定时间 $t$ 能否到达包围圈的确定

设  $C(t)$  中节点个数为  $L$ ，可分配的平台为  $K = 80$  个。我们的目标是采用问题 2 中的模型对平台的警力进行分配，求出到达各节点的最长时间  $T(t)$ 。 $T(t)$  为模型中 (a) 的最优值。

$$\min Z = \max_j T_j$$



$$s.t. \begin{cases} \sum_{j=1}^L x_{ij} \leq 1 & i=1,2,\dots,80 \\ \sum_{i=1}^{80} x_{ij} = 1 & j=1,2,\dots,L \\ T_j = \sum_{i=1}^{80} d_{ij} x_{ij} / v \\ x_{ij} = 0 \text{ 或 } 1 \end{cases}$$

若  $T(t) \leq t-3$ ，则警察可以完成包围圈  $C(t)$  的包围。否则不能完成包围。 $t-3$  是因为罪犯先逃跑了 3min。

我们的目标就是求出在最短时间完成对罪犯的包围。因此目标函数为：

$$\min Z = t$$

约束为  $T(t) \leq t-3$ 。

因此总的模型为：

$$\min Z = t$$

$$s.t. \quad T(t) \leq t-3$$

实际计算时，可以采用从  $t=3$  开始，每隔 30 秒进行计算，直到满足约束条件  $T(t) \leq t-3$  为止。

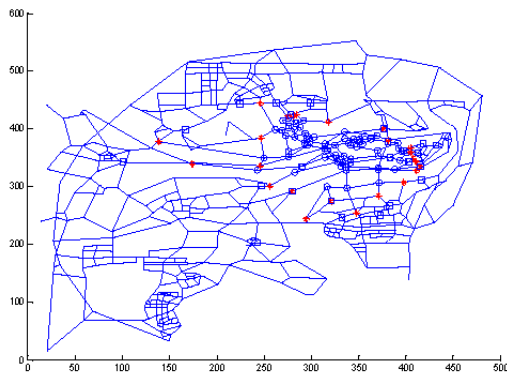
实际计算中平台数可以减少。若某平台的警力在  $t-3 \text{ min}$  内都不能到达包围圈  $C_t$  中任意点，则该平台删除。

即可选平台集合为：

$$\text{Plat} = \{i \mid \text{Time}(i, j) < t-3, \exists j \in C_t, i=1,2,\dots,80\}$$

这样可大大减少平台数，使问题规模减小，便于求解。

计算程序见附件 5 中的 b2011\_5\_1.m 和附件 6 中的 b2011\_5\_2.m。计算得到的结果见围堵方案图 5-19 和围堵方案表 5-13 所示。



其中：○为内部点，\*为包围点，□为分配的平台。

图 5-19



表 5-13

序 号	包 围 点	平 台 号	封锁时间 (min)	罪犯逃跑到该点时间 (min)
1	14	14	3	10.0432
2	17	17	3	10.6504
3	26	11	3.9	9.72654
4	29	15	8.7005	9.15563
5	41	2	6.4411	10.5025
6	43	1	4.8001	9.38794
7	62	4	3.35	9.1319
8	68	19	5.2384	9.15307
9	70	3	5.9719	9.44818
10	76	18	6.1328	9.23845
11	168	168	3	12.4791
12	215	171	6.6274	11.2542
13	217	172	3.9717	9.83724
14	218	170	6.2789	9.44592
15	227	174	6.6998	9.22363
16	240	173	9.6952	10.1541
17	248	167	6.6788	20.5237
18	273	182	5.1024	12.8088
19	371	320	10.361	15.894
20	482	482	3	11.1731
21	487	481	7.565	14.6402
22	549	476	6.0255	10.7206
23	558	475	5.0526	11.0991
24	562	480	4.9416	11.3988

警察在接到报案后，最长完成封锁时间为 $10.361 - 3 = 7.361 \text{ min}$ 。

5.2.3 点评

在完成该题过程中，需要用 MatLab 编程与 Lingo 编程相结合共同完成问题求解。利用 MatLab 可以进行数据预处理，根据 Floyd 算法进行最短路计算，对包围路线作图，以及对自己设计的算法进行计算。Lingo 则可以直接根据模型进行优化计算，而其中使用的大量数据需要通过 MatLab 计算输出。这种通过 MatLab 和 Lingo 相互结合完成计算，是近年来在竞赛中经常使用的方法。

5.2.4 程序附件

附件 1：数据预处理及输出数据文件的 MatLab 程序 b2011\_1.m

```
%初始数据处理,作图,并得到 92 个路口之间的最短路 A(92,92)
%计算并输出 20 个平台与 13 个路口之间距离 dis1.txt
```



```
%输出 20 个路口到待分配路口(86 个)的距离矩阵(20×86) dt1.txt
%输出待分配路口的(86)口的发案数 ft1.txt
%还可对 LINGO 计算结果按照规范形式输出。
load DA.txt %所有路口号(582 个), x,y,区,发案率
%街道数据
load DB.txt; %连接线,928 条
DA=DA(1:92,:); %取前 92 个路口信息
[ma,na]=size(DA);
[mb,nb]=size(DB);
pos=[12,14,16,21,22,23,24,28,29,30,38,48,62];%A 区出入口位置
k=20; %1--20 平台
x=DA(:,2); y=DA(:,3); f=DA(:,5);%分别获得个路口的 x 坐标, y 坐标,发案数
plot(x,y,'bo',x(1:k),y(1:k),'g*'); %作图,蓝色为所有路口,绿色为前 20 个平台
kp=0; %统计 A 区街道数
LineA=zeros(200,2);
for i=1:mb
    if DB(i,1)<=92&&DB(i,2)<=92
        kp=kp+1;
        LineA(kp,1)=DB(i,1);
        LineA(kp,2)=DB(i,2);
    end
end %将 A 区街道的起始点和终点分别放在数组 LineA(kp,1)和 LineA(kp,2)中。

hold on
fprintf('A 区街道数%d\n',kp);
px=zeros(2,1); py=zeros(2,1);
for i=1:kp
    px(1)=x(LineA(i,1)); py(1)=y(LineA(i,1));
    px(2)=x(LineA(i,2)); py(2)=y(LineA(i,2));
    plot(px,py); %将 A 区所有街道连线
end
%利用 Floyd 算法求任意两点之间最短路
v=1000; %每分钟速度
n=92; %A 区路口数为 92
A=zeros(n,n); %存储任意两点最短路
for i=1:n
    for j=1:n
        if(i==j) A(i,j)=0;
        else A(i,j)=1000000;
        end
    end
end
end

for i=1:kp %给每条线路赋予距离
    point1=LineA(i,1); point2=LineA(i,2);
    plx=x(point1); ply=y(point1);
    p2x=x(point2); p2y=y(point2);
    d=sqrt((plx-p2x)^2+(ply-p2y)^2);
```

```

A(point1,point2)=d;
A(point2,point1)=d;
end
U=A; %原始矩阵
%1.1 利用 Floyd 算法计算最短距离矩阵
for k=1:n
for i=1 :n
for j=1:n
t=A(i,k)+A(k,j);
if t<A(i,j) A(i,j)=t; end
end %end for j
end %end for i
end %end for k

A=100*A; %每毫米代表 100 米,得到以米为单位的距离
Dis=zeros(20,13); %存储 20 个平台和 13 个交通要道的距离
for i=1:20
for j=1:13
ip=i; %第 ip 个平台
jp=pos(j); %第 jp 个路口
Dis(i,j)=A(ip,jp);
end
end %获得 20 个平台和 13 个交通要道(路口)的距离矩阵

%输出 20 个平台与 13 个路口之间距离
fid=fopen('d:\lingo12\dat\dis1.txt','w'); %此处目录可修改
for i=1:20
for j=1:13
fprintf(fid,'%6.2f ',Dis(i,j));
end
fprintf(fid,'\n');
end
fclose(fid);

n=92; %路口数
m=20; %平台数
Plat=1:m; %平台

T=zeros(1,n); %存储到各路口的时间
S=zeros(1,n); %存储到各路口时间最少的平台号
%求到各路口的最短时间和对应的平台号
for j=1:n
jp=j; %路口
mint=1000;
for i=1:m
ip=Plat(i); %平台
t=A(ip,jp)/v;
if t<mint mint=t; xu=ip; end

```





```
end

T(j)=mint; %到第 j 个路口的最短时间
S(j)=xu; %到第 j 个路口的时间最短的平台号
end

%计算各路口到平台的最短时间
fprintf('到平台超过 3 分钟的路口信息.\n');
fprintf('路口, 平台, 最短时间, 任务量\n');
U=ones(1,n);
for j=1:n
    if T(j)>3.0 U(j)=0;
    fprintf('%2d %2d %6.2f %6.2f\n',j,S(j),T(j),f(j)); end
end %输出超过 3 分钟到达的路口
%获得待分配的路口序号 (时间不超过 3 分钟的路口)
Lu=[];
for j=1:n
    if U(j)==1 Lu=[Lu,j]; end
end
r=length(Lu); %剩余路口数
%获得 20 个平台到剩余路口 (86) 的距离矩阵
D=zeros(m,r);%(平台, 路口)
for i=1:m %平台数 m=20
    for j=1:r %剩余路口数 r=86
        ip=Plat(i);
        jp=Lu(j);
        D(i,j)=A(ip,jp);
    end
end
end

fid=fopen('d:\lingo12\dat\dt1.txt','w'); %输出 20×86 的距离矩阵
for i=1:m
    for j=1:r
        fprintf(fid,'%6.2f ',D(i,j));
    end
    fprintf(fid,'\n');
end
fclose(fid);

fid=fopen('d:\lingo12\dat\ft1.txt','w'); %输出剩余 86 个路口的任务量
for j=1:r
    k=Lu(j);
    fprintf(fid,'%5.1f\n', f(k));
end
fclose(fid);

%此段程序对 LINGO 程序 b2011_1.lg4 计算结果进行整理, 开始时不需要。
TEST=1; %当 TEST=1, 对 LINGO 计算结果进行规范输出, 便于阅读和写作。
```

```

if TEST==1
    %对对 Lingo 求解结果进行整合，重新按照规范格式输出。
    x=zeros(m,r);
    w=[0,1.4,0,0,0,0,0.6,0,0,0,0,0,0,2.7,1.2,0,0,0,0.8]; %初始各平台的任务量
    %将 LINGO 计算结果拷贝在下面：
    ;x(1,1)=1;x(1,59)=1;x(1,66)=1;x(1,67)=1;x(1,72)=1;x(1,73)=1;
    x(1,74)=1;x(1,75)=1;x(2,2)=1;x(2,36)=1;x(2,64)=1;x(2,69)=1;
    x(3,3)=1;x(3,40)=1;x(3,50)=1;x(3,51)=1;x(3,65)=1;x(3,71)=1;
    x(4,4)=1;x(4,53)=1;x(4,56)=1;x(4,57)=1;x(4,58)=1;x(4,60)=1;
    x(4,61)=1;x(5,5)=1;x(5,45)=1;x(5,49)=1;x(5,52)=1;x(5,55)=1;
    x(6,6)=1;x(6,46)=1;x(6,47)=1;x(6,48)=1;x(6,54)=1;x(7,7)=1;
    x(7,28)=1;x(7,44)=1;x(8,8)=1;x(8,30)=1;x(8,31)=1;x(8,43)=1;
    x(9,9)=1;x(9,33)=1;x(9,34)=1;x(9,35)=1;x(9,41)=1;x(10,10)=1;
    x(11,11)=1;x(11,26)=1;x(11,27)=1;x(12,12)=1;x(12,25)=1;x(13,13)=1;
    x(13,21)=1;x(13,22)=1;x(13,23)=1;x(13,24)=1;x(14,14)=1;x(15,15)=1;
    x(15,29)=1;x(16,16)=1;x(16,32)=1;x(16,42)=1;x(17,17)=1;x(17,37)=1;
    x(17,38)=1;x(17,39)=1;x(18,18)=1;x(18,80)=1;x(18,82)=1;x(18,84)=1;
    x(18,85)=1;x(18,86)=1;x(19,19)=1;x(19,62)=1;x(19,63)=1;x(19,68)=1;
    x(19,70)=1;x(19,77)=1;x(19,78)=1;x(20,20)=1;x(20,76)=1;x(20,79)=1;
    x(20,81)=1;x(20,83)=1;

    %输出每个平台管辖的路口，任务量,最长时间
    fprintf('\n 第一问 LINGO 求解结果的输出.\n');
    fprintf('平台，管理的路口，任务量，最长时间:\n');
    TT=zeros(m,1);
    W=zeros(m,1);
    for i=1:m
        fprintf('%2d:',i);
        s1=w(i); %初始任务
        TT(i)=0;
        for j=1:r
            if x(i,j)==1
                jp=Lu(j);
                s1=s1+f(jp);
                t=A(i,jp)/v;
                if t>TT(i) TT(i)=t;end
                fprintf('%2d,',jp);
            end
            W(i)=s1;
        end
        fprintf(' w=%5.2f t=%5.2f\n',s1,TT(i));
    end
end

```

## 附件 2：问题 1 的 Lingo12 程序 b2011\_1.lg4

该程序对 86 个待分配路口进行分派，所使用的数据文件 dt1.txt 和 ft1.txt 由 MatLab 程



序 b2011\_1.m 生成。Lingo 程序的计算结果可直接拷贝到 b2011\_1.m 中最后相应位置，重新按照规范格式输出。在 b2011\_1.m 中设置 TEST=1 控制输出。

```
! 问题 1 的 LINGO 程序;
model:
sets:
Plat/1..20/:s,w;
Kou/1..86/:T,f;
Assign(Plat,Kou):dis,x;
endsets
data:
v=1000;
n=20;
dis=@file('d:\lingo12\dat\dt1.txt'); !20 个平台和待分配的 86 个路口的距离矩阵;
f=@file('d:\lingo12\dat\ft1.txt'); ! 待分配的 86 个路口的发案量;
@text()=@writefor(Assign(i,j)|x(i,j)#GT#0:'x(' ,i , ',' ,j ,')=' ,x(i,j) , '
');
w=0,1.4,0,0,0,0,0.6,0,0,0,0,0,0,0,2.7,1.2,0,0,0,0.8;!20 个平台的初始发案量;
enddata

min=@sqrt(@sum(Plat(i):(s(i)-sv)^2)/(n-1)); !标准差最小;
@for(Kou(j):@sum(Plat(i):x(i,j))=1); !每个路口恰好有一个平台;
@for(Plat(i):@sum(Kou(j):x(i,j))>=1);!每个平台至少管理一个路口;
@for(Kou(j):T(j)=@sum(Plat(i):x(i,j)*dis(i,j)/v)); !第 j 个路口的处理时间;
@for(Kou(j):T(j)<=3); !每个路口到达时间少于 3 分钟;
@for(Plat(i):x(i,i)=1); !每个平台管理自己的路口;
@for(Plat(i):s(i)=w(i)+@sum(Kou(j):f(j)*x(i,j)));
!计算各平台管辖的路口的任务量;
sv=@sum(Plat(i):s(i))/n; !平均路口数;
@for(Assign(i,j):@bin(x(i,j)));
end
```

### 附件 3: 问题 2 的 Lingo12 程序 b2011\_2.lg4

该程序对 13 个交通要道的围堵问题优化求解。

```
model:
sets:
Plat/1..20/;
Kou/1..13/:T;
Assign(Plat,Kou):dis,x,Time;
endsets
data:
v=1000;
dis=@file('d:\lingo12\dat\dis1.txt');!20 个平台和 13 个路口的距离矩阵;
@text()=@writefor(Assign(i,j)|x(i,j)#GT#0:'x(' ,i , ',' ,j ,')=' ,x(i,j) , ' ');
enddata
```

```

min=aver;
! min=TT;
TT=@max(Assign(i,j):Time(i,j)); !最大时间最小化;
aver=@sum(Kou(j):T(j))/13;!平均时间;
TT<8.02; !当优化第二目标时将第一目标变为约束;
@for(Assign(i,j):Time(i,j)=x(i,j)*dis(i,j)/v);
@for(kou(j):T(j)=@sum(Plat(i):x(i,j)*dis(i,j))/v);!到第j个路口的时间;
@for(Plat(i):@sum(Kou(j):x(i,j))<=1); !第i个平台最多到一个路口;
@for(Kou(j):@sum(Plat(i):x(i,j))=1); !第j个路口一定有一个平台到达;
@for(Assign(i,j):@bin(x(i,j)));
end

```

#### 附件 4: b2011 问题 3 新增平台的算法程序 b2011\_3.m

```

%A(582,582)任意两点之间的最短距离,先运行 b2011_5_1.m 获得最短距离矩阵 A(582,582)
%采用单选法,每次从超过 3 分钟的路口中选取 1 个降低最长时间最多的路口
%考察 B 区
Qu=1; %1(A 区);2(B 区),3(C 区),4(D 区),5(E 区),6(F 区)
%输出不同区号可计算不同区增加平台情形.

load DC.txt; %平台号(80 个),所属区
K=length(DC);
Plat=[]; %存储某区平台
n=0; %统计某区已有平台数
for i=1:K
    Belong=DC(i,2); %1(A 区);2(B 区),3(C 区),4(D 区),5(E 区),6(F 区)
    if Belong==Qu Plat=[Plat,DC(i,1)]; n=n+1;end
end
load DA.txt; %路口号(92 个),x,y,区,发案率
Lu=[]; % 存储某区路口数
fz=[]; %存储发案率

L=length(DA);

m=0; %统计某区已有路口数
for i=1:L
    Belong=DA(i,4); %路口号(92 个),x,y,区,发案率
    if Belong==Qu Lu=[Lu,DA(i,1)]; fz=[fz,DA(i,5)]; m=m+1;end
end

%计算各路口到各平台的最短时间
%1. 初始情形:
x=zeros(1,m);%确定每个路口受管辖的平台
T=zeros(1,m);%每个路口到所管辖平台的时间
v=1000; %速度

for i=1:m
    tmin=100;

```





```
for j=1:n
    ip=Lu(i); %路口号
    jp=Plat(j); %平台号
    t=A(ip,jp)/v;
    if t<tmin tmin=t; pj=jp; end
end
x(i)=pj; %第 i 个路口受管辖的平台
T(i)=tmin; %第 i 个路口到所管辖平台的时间

end

fprintf('路口,平台,时间:\n');

for i=1:m
    fprintf('%2d %2d %5.2f\n',Lu(i),x(i),T(i));
end

%获得超过 3 分钟的路口,平台,时间
S=zeros(40,3);
p=0;
for i=1:m
    if T(i)>3
        p=p+1;
        S(p,1)=i; S(p,2)=x(i); S(p,3)=T(i);
    end
end
fprintf('区号%2d, 路口数%2d, 平台数%2d, 超过 3 分钟的路口%2d\n',Qu,m,n,p);
fprintf('超过 3 分钟的路口,平台,时间\n');
for i=1:p
    fprintf('%2d %2d %5.2f\n',Lu(S(i,1)),S(i,2),S(i,3));
end

maxT=max(T); %最长时间

fprintf('增加平台数,最长时间\n');
%方法,每次从超过 3 分钟的路口中选取 1 个降低最长时间最多的路口

Temp_Platform=zeros(1,m);
Have_Platform=zeros(1,m);
Yu=zeros(1,m); %剩余平台
for i=n+1:m
    Yu(i)=1;
end %获得当前剩余的 platform
for i=1:n
    Temp_Platform(i)=Plat(i);
    Have_Platform(i)=Plat(i); %已经选定的 platform
end

for Kp=1:22 %增加 platform, 22 为最多增加数量
```



```

flag=0;

for k=1:p
    sel=S(k,1); %选取 1 个超过 3 分钟的平台序号
    Temp_Platform(n+Kp)=Lu(sel);

    for i=1:m
        tmin=100;
        for j=1:n+Kp %平台数
            ip=Lu(i);
            jp=Temp_Platform(j);
            t=A(ip,jp)/v;
            if t<tmin tmin=t; pj=j; end
        end %end for j

        x(i)=pj; %第 i 个路口受管辖的平台
        T(i)=tmin; %第 i 个路口到所管辖平台的时间

        end %end for i

        maxT1=max(T);
        if maxT1<maxT maxT=maxT1; Add_Platform=sel; flag=1; end

    end %end for k

    if flag==1 %Add_Platform 为平台在路口中的序号
        Temp_Platform(n+Kp)=Lu(Add_Platform); Have_Platform(n+Kp)=Lu(Add_Platform);
        Yu(Add_Platform)=0;
        flag=0;

        for i=1:m
            tmin=100;
            for j=1:n+Kp
                ip=Lu(i);
                jp=Have_Platform(j);
                t=A(ip,jp)/v;
                if t<tmin tmin=t; pj=j; end
            end %end for j
            x(i)=pj; %第 i 个路口受管辖的平台
            T(i)=tmin; %第 i 个路口到所管辖平台的时间
        end %end for i

        fprintf('Kp=%2d Platform=%2d maxT=%5.4f\n',Kp,Lu(Add_Platform),max(T));
        end %end if flag==1
    end %end for Kp

```

当对 A 区计算, 结果为:

增加平台数, 最长时间

Kp= 1 Platform=28 maxT=4.1902

Kp= 2 Platform=61 maxT=3.6822





```
Kp= 3  Plat=38  maxT=3.6013
Kp= 4  Plat=92  maxT=2.7083
```

附件 5 求取最短距离矩阵  $A(582,582)$  的程序 b2011\_5\_1.m

```
clear;
load DA.txt; %路口号(92个), x,y,区,发案率
load DB.txt; %连接线 928 条
load DC.txt; %平台号, 80 个
load DD.txt; %全市出入市区路口数(17个)
x=DA(:,2); y=DA(:,3);
b=DA(:,4); %所属区 s
f=DA(:,5); %发案率
[ma,na]=size(DA);
[mb,nb]=size(DB);
kp=length(DB); %连接线总数
px=zeros(2,1); py=zeros(2,1);
hold on
for i=1:kp
    px(1)=x(DB(i,1)); py(1)=y(DB(i,1)) ;
    px(2)=x(DB(i,2)); py(2)=y(DB(i,2)) ;
    %sprintf('%5.0f,%5.0f--(%5.0f,%5.0f)\n',px(1),py(1),px(2),py(2));
    plot(px,py); %画原始点及线路
end

%形成线的 0~1 邻接矩阵 P
P=zeros(kp,kp);
for i=1:kp
    i1=DB(i,1); j1=DB(i,2);
    P(i1,j1)=1;
    P(j1,i1)=1;
end
pos=32; %罪犯逃跑点
plot(x(pos),y(pos),'ro');
%利用 Floyd 算法求任意两点之间最短路
v1=1000; %罪犯逃跑速度
n=ma; %582
A=zeros(n,n); %存储任意两点最短路
for i=1:n
    for j=1:n
        if(i==j) A(i,j)=0;
        else A(i,j)=1000000;
        end
    end
end
end
for i=1:kp %给每条线路赋予距离
    point1=DB(i,1); point2=DB(i,2);
    p1x=x(point1); p1y=y(point1) ;
    p2x=x(point2); p2y=y(point2);
```

```

d=sqrt((p1x-p2x)^2+(p1y-p2y)^2);
A(point1,point2)=d;
A(point2,point1)=d;
end

U=A; %原始矩阵
%1.1 利用 Floyd 算法计算最短距离矩阵
for k=1:n
for i=1 :n

    for j=1:n
        t=A(i,k)+A(k,j);
        if t<A(i,j) A(i,j)=t; end
    end %end for j
end %end for i
end %end for k
A=100*A; %每毫米代表 100 米
%获得 A(582,582)为任意两点之间的最短距离

```

#### 附件 6: 完成围堵的程序 b2011\_5\_2.m

```

%先运行 b2011_5_1.m 求出任意两点之间的最短距离 A(582,582)
%求给定 t 时的集合 At,Bt,Ct,Pt(可选平台集合)
%给定时间 t 时
v=1000; %罪犯速度

t=9.1; %给定时间
At=[];
num1=0;
%step1 求 t 分钟内形成的范围 At
for i=1:582
    if A(i,pos)/v1<t
        num1=num1+1;
        At=[At,i];
    end
end

hold on
for i=1:num1
    plot(x(At(i)),y(At(i)),'bo'); %画 At 点
end

%step2 求与 At 相邻的点 Bt
Bt=[];
U=zeros(582,1);
for i=1:num1
    i1=At(i);
    for j=1:582
        if P(i1,j)==1 U(j)=1; end %找出与 At(i)相连的点
    end
end

```





```
end
end

for i=1:582
    if U(i)==1 Bt=[Bt,i]; end
end
num2=length(Bt);
%step3 求 C(t)=B(t)-A(t)
U=zeros(582,1);

for i=1:num1
    k=At(i);
    U(k)=1;
end
Ct=[];
for i=1:num2
    i1=Bt(i);
    if U(i1)==0 Ct=[Ct,i1]; end
end

kp=length(DB); %连接线总数
px=zeros(2,1); py=zeros(2,1);
hold on

for i=1:kp
    px(1)=x(DB(i,1)); py(1)=y(DB(i,1));
    px(2)=x(DB(i,2)); py(2)=y(DB(i,2));
    plot(px,py); %画原始点及线路
end

pos=32; %罪犯逃跑点
plot(x(pos),y(pos),'bo');
%求到包围点最近的平台
kp=length(DC);
U=zeros(kp,1);
Dp=zeros(length(Ct),1); %包围点获得指定的平台
TT=zeros(length(Ct),1); %存储平台到包围点的时间
for i=1:length(Ct);
    tmin=1000;
    for j=1:length(DC);
        tt=A(Ct(i),DC(j,1))/v;
        if tt<tmin &&U(j)==0 s=j; tmin=tt; end
    end
    TT(i)=tmin; U(s)=1; Dp(i)=DC(s,1);
end
for i=1:length(Ct);
    T1(i)=A(Dp(i),Ct(i))/v;
    T2(i)=A(pos,Ct(i))/v;
end
for i=1:length(Ct)
```

```

plot(x(Ct(i)),y(Ct(i)),'ro'); %画包围圈点
end
for i=1:length(Dp)
plot(x(Dp(i)),y(Dp(i)),'bs'); %画平台点
end
fprintf('序号 新包围点,平台,封锁时间,罪犯到达时间\n');
for i=1:length(Ct)

fprintf('%4d %4d %4d %6.4f %6.4f\n',i,Ct(i),Dp(i),TT(i)+3,T2(i));
end
Info=zeros(length(Ct),4);
for i=1:length(Ct)
Info(i,1)=i;Info(i,2)=Ct(i); Info(i,3)=Dp(i); Info(i,4)=TT(i)+3; Info
(i,5)=T2(i);
end
XLSWRITE('rest.xls.',Info,'A1:E24'); %输出到XLS文件中.

```

## 5.3 2014 MCM-A

### 5.3.1 引言

#### 一、问题综述

“2014 年美国大学生数学建模竞赛 A 题《除非超车否则靠右行驶的交通规则》”是典型的交通问题。有关交通方面的问题在近些年来的数学建模比赛中经常出现。这类问题大多数都是要求参赛者在宏观上探讨车流流量，流速和密度之间的关系，微观上研究车与车之间的相互作用及不同外界环境（路段，信号，规则）的影响，以求减少交通时间的延误，事故的发生和提高道路交通设施通行<sup>[1]</sup>。以下列出了近 10 年来数学建模比赛中出现过的涉及交通的问题。

- 2005 年美赛 B 题：Tollbooths<sup>[2]</sup>
- 2009 年美赛 A 题：Designing a Traffic Circle<sup>[3]</sup>
- 2013 年国赛 A 题：车道被占用对城市道路通行能力的影响<sup>[4]</sup>
- 2014 年美赛 A 题：The Keep-Right-Except-To-Pass Rule<sup>[5]</sup>

由于交通问题存在很多共性，因此对于 2014 年的美赛，参加过 2013 年国赛并且做 A 题《车道被占用对城市道路通行能力的影响》的参赛者会有少许优势。至少在前期的调研，基本交通理论的学习，以及模拟方法的选择和实现方面要轻松得多。有关交通方面的数学建模问题，常用的交通流模型主要有宏观模型和微观模型<sup>[6,7]</sup>，宏观模型将交通流比拟为连续的流体，类比于流体力学的相关理论。微观模型则运用动力学方法从微观角度研究车辆列队在无法超车的单一车道上的行驶状态和规律<sup>[7]</sup>。在数学建模比赛中，微观模型更为常用，经常被用来模拟不同的交通问题。





本文主要对交通问题常用微观模型，特别是多车道下换道模型做基本的介绍及展示。并以 2014 美赛 A 题的一篇特等奖论文为例作简要分析。

2014 年的美赛 A 题是涉及交通方面的问题，要求考虑在多车道情况下，分析靠右行驶规则在交通流畅时和在交通拥挤的情况下的性能表现（交通流量及安全性），并与改进的交通规则或靠左行驶规则做对比，最终还要求参赛者在模型基础上，提出一种不依赖人判断的智能控制系统。赛题原文<sup>[5]</sup>如下：

#### 2014 MCM Problem A: The Keep-Right-Except-To-Pass Rule

In countries where driving automobiles on the right is the rule (that is, USA, China and most other countries except for Great Britain, Australia, and some former British colonies), multilane freeways often employ a rule that requires drivers to drive in the right-most lane unless they are passing another vehicle, in which case they move one lane to the left, pass, and return to their former travel lane.

Build and analyze a mathematical model to analyze the performance of this rule in light and heavy traffic. You may wish to examine tradeoffs between traffic flow and safety, the role of under- or over-posted speed limits (that is, speed limits that are too low or too high), and/or other factors that may not be explicitly called out in this problem statement. Is this rule effective in promoting better traffic flow? If not, suggest and analyze alternatives (to include possibly no rule of this kind at all) that might promote greater traffic flow, safety, and/or other factors that you deem important.

In countries where driving automobiles on the left is the norm, argue whether or not your solution can be carried over with a simple change of orientation, or would additional requirements be needed.

Lastly, the rule as stated above relies upon human judgment for compliance. If vehicle transportation on the same roadway was fully under the control of an intelligent system – either part of the road network or imbedded in the design of all vehicles using the roadway – to what extent would this change the results of your earlier analysis?

在 2014 年的美赛中共计有 6755 个参赛队成功提交了 MCM 参赛作品，其中 A 题有 6 个队获得了特等奖。这 6 个队分别来自上海交通大学，清华大学，南京大学，浙江大学，北京师范大学和 Tufts University。值得一提的是，6 篇特等奖论文全部采用了元胞自动机模型。

## 二、交通流的元胞自动机模型

元胞自动机模型及单车道 NS 元胞自动机模型在本书的和 4 章中已经做了较为详细的介绍，这里不再重复。基于单车道 NS 元胞自动机模型，加入适当的换道规则，就可以模拟双车道或多车道情况。本节来讨论双车道情况，在多车道 NS 模型中，各条车道上行驶



的车辆遵守 NS 规则, 在进行车道变改时满足变道规则。很多学者提出了不同的换道规则<sup>[8,9]</sup>, 大部分规则都是从左往右和从右往左同时进行的, 但也有个别文献的规则不同, 比如文献<sup>[8]</sup>中提出的换道规则是偶数步执行从右往左换, 奇数步执行从左往右换。本文以 Nagel 等<sup>[10]</sup>提出的右行规则为例进行介绍。具体规则如图 5-20 所示 (以红车为例):

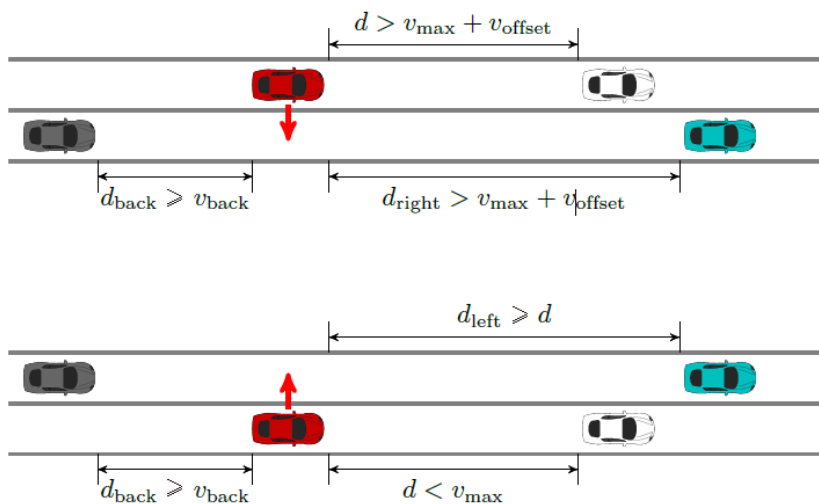


图 5-20

- 从右往左换: 如果右车道上的红车到前方车辆的间距  $d < v_{\max}$ , 并且对于左车道上前方的车辆间距  $d_{\text{left}} \geq d$ , 则此时红车想要换到左车道。
- 从左往右换: 从左往右换与从右往左换稍有不同 (如果相同, 则是两车道等价的情况), 如果处于左车道的红车满足  $d > v_{\max} + v_{\text{offset}}$ ,  $d_{\text{right}} > v_{\max} + v_{\text{offset}}$ , 则此时红车想要换到右车道。 $v_{\text{offset}}$  是一个控制参数, 用来控制非对称换道规则的。具体引入方式见文献<sup>[10,9]</sup>。
- 为保证换车道后, 不造成后方车辆的追尾, 还需保证  $d_{\text{back}} \geq v_{\text{back}}$ 。在此前提下以一定概率 ( $p_{\text{l2r}}$  及  $p_{\text{r2l}}$ ) 换道。

通过简单的分析, 不难发现以上规则具有一定的限制性, 对于密度比较大的情况下, 大部分车量都集中在左车道, 右车道甚至可能出现没有车的情况。

对于左右道等同的情况, 比较简单, 只需将从左往右换的规则改成与从右往左换的规则相同即可, 比赛中最好对这种情况也模拟一下, 这样右行规则才有对比的参照。对于三车道及更多车道的交通规则, 类似双车道, 可以参考文献<sup>[11]</sup>。

接下来本文应用 NS 规则及上述右行换道规则对双车道进行模拟。在一条长为  $L = 100\text{m}$  的封闭环形车道上 (相当于周期性直行车道), 改变车流密度  $\rho \in [0, 1]$ , 在每种密度的情况下进行多次模拟, 测试不同密度情况下的平均流量和平均速度。车辆的最大速度  $v_{\max} = 5 \text{ m/s}$ , 为消除随机因素, 测试中本文将随机减速的概率设为  $v_{\text{brake}} = 0$ , 换道概率  $p_{\text{l2r}} = p_{\text{r2l}} = 1$ 。

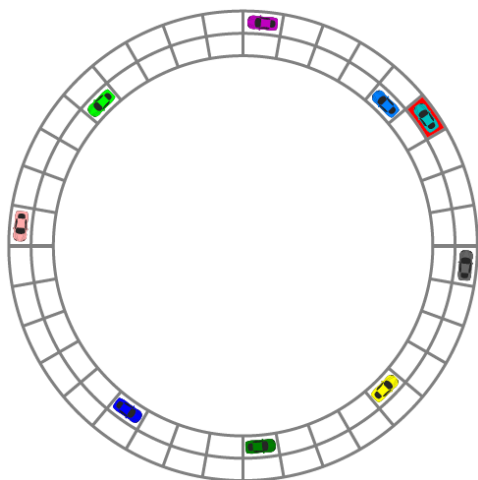


图 5-21

图 5-21 为单车道 NS 规则模拟的效果图（为了出图效果，效果图中的道路长度被降为 36m）。图中右道红色格子中的车是静止的，以模拟某车道被占用的情况。值得注意的是，车道被占用正是 2013 年国赛 A 题。在模拟中，其他车辆会通过换行到左道来绕过右车道上的故障车辆，绕过故障车辆后，如果满足回到右道的条件时又会换行到右道。然而经过更多的测试，可发现一些现象：

- 当密度小于约 0.2 时，左道上几乎没有车。这是合理的，密度比较小时，所有车都是自由行驶，无需换道。
- 对于密度大于 0.7 时，几乎所有车量都集中在左车道，右车道出现了没有车的情况。这显然与实际情况不合。这是由这个模型本身的限制造成的。

以上第二种情况的缺陷，可以通过简单的改进模型来消除：在模型中加入一条规则，如果右车道有足够的空间的话，左车就以一定的概率换回到右车道上。并将  $p_{l2r}$  设置的远小于  $p_{r2l}$ ，比如文献<sup>[9]</sup>中设置  $p_{l2r} = 0.05$ ， $p_{r2l} = 0.2$ 。有关双车道右行规则模拟得到的结果（密度-流量图，密度-速度图……），这里不再一一展示。

单车道 NS 元胞自动机模型的 MatLab 实现方法在本书的第 4 章已经做了详细的介绍。在此基础上，双环形车道交通流的元胞自动机 NS 模型也比较容易实现，可以定义如下函数来实现 NS 模型。相关参数比如平均流量，可以参照单车道 NS 元胞自动机模型的计算方法适当添加代码。下面给出双环形车道交通流的元胞自动机 NS 主程序。

```

1 function ns(rho , p, L, tmax , pchange)
2 % rho = 0.15; p = 0.25; L = 100; tmax = 5000; pchange = 0.5;
3
4 vmax = 5; % 车辆最大行驶速度
5 ncar = round(L*2* rho); % 根据密度  $\rho$  计算系统中车辆数  $ncar = [L \cdot \rho]$ 
6 rho = ncar / 2 / L;
7
8 xy = randperm (2*L,ncar);

```



```

9  [y,x] = ind2sub ([2, L], xy); % 车量的位置. y 表示车道: 1 左, 2 右.
10 v = vmax * ones(1,ncar); % 速度向量: 初始化所有车辆的速度为 vmax
11
12 voffset = 1; % 换道参数
13 vback = 1; % 给后方车辆的预留速度
14
15 for t = 1:tmax
16     [gaps , gapfront , gapback] = gaplength(x,y,L);
17
18     % 左换右& 右换左
19     l2r = find(y==1 & gaps >vmax+voffset & gapfront >vmax+voffset &...
20     gapback >= vback);
21     r2l = find(y==2 & gaps <vmax & gapfront >gaps &...
22     gapback >= vback & rand(size(y))<pchange);
23     y(l2r) = 2;
24     y(r2l) = 1;
25
26     v = min(v+1, vmax); % 加速规则
27
28     gaps = gaplength(x,y,L);
29
30     v = min(v, gaps -1); % 防止碰撞
31
32     vdrops = ( rand(1,ncar)<p );
33     v = max(v-vdrops ,0); % 随机减速
34
35     x = x + v; % 位置更新
36     x(x>L) = x(x>L) - L; % 周期边界
37 end

```

在结构上,以上程序与 4.3 节中单车道 NS 元胞自动机模型的 MatLab 程序基本相同。需要说明的是

- 第 16 行: 车辆之间距离的函数 `gaplength` 返回了三个参数: `gap` 为与同车道上前车距离; `gapfront` 为与另一车道前车距离; `gapback` 为与另一车道后车距离。
- 第 19 行: 左车道上满足条件的车换到右车道, 对应图 5-20 中的上图。类似的, 第 21 行为右车道换到左车道, 对应图 5-20 中的下图。

以上 NS 主程序还调用了函数 `gaplength` 来分别计算当前车辆与同车道上前车距离, 与另一车道前车距离和与另一车道后车距离。函数 `gaplength` 的代码如下:

```

1  function [gap , gapfront , gapback] = gaplength(x,y,L)
2  % 输入: x- 位置; y- 车道; L- 环形连道长度
3  % 输出: gap - 与同车道上前车距离;
4  % gapfront - 与另一车道前车距离; gapback - 与另一车道后车距离
5
6  ncar = length(x);
7
8  % 初始化每辆车的 gap, gapfront, gapback 为无穷大

```





```

9   gap = inf*ones(1, ncar);
10  gapfront = inf*ones(1, ncar);
11  gapback = inf*ones(1, ncar)
12
13  index = 1:ncar;
14  for i = index
15      j1 = index(index~=i&y==y(i)); % 找出和自己在同一车道上的车
16      if ~isempty(j1) % 是否存在这样的车
17          d1 = x(j1) - x(i); % 计算与前车距离
18          d1(d1 < 0) = d1(d1 < 0) + L; % 周期边界条件
19          gap(i) = min(d1); % 找出最近的距离
20      end
21
22      j2 = index(index ~= i&y ~= y(i)); % 找出和自己不在同一车道上的车
23      if ~isempty(j2) % 是否存在这样的车
24          d2 = x(j2) - x(i); % 计算与前车距离
25          d2(d2 < 0) = d2(d2 < 0) + L; % 周期边界条件
26          gapfront(i) = min(d2); % 找出最近的距离
27
28          d3 = x(i) - x(j2); % 计算与后车距离
29          d3(d3 < 0) = d3(d3 < 0) + L; % 周期边界条件
30          gapback(i) = min(d3); % 找出最近的距离
31      end
32  end

```

### 三、交通流的粒子方法模型

虽然元胞自动机方法非常简单，并且能一定程度上很好的模拟交通问题。但元胞自动机模拟交通流存在很多缺点，比如空间离散，时间离散带来的不确定性，每个格子与真实对应多大的物理空间？每个时间步对应多长的物理时间？这些都不是很容易精确的。而且也很难很好地表示不同尺寸，不同速度的车辆。因此本节中，再介绍一种基于粒子方法模拟实现单车道和多车道的交通流模拟。粒子方法把车辆看成一个粒子（有尺寸），在一维或二维的空间上连续的运动。本文主要介绍 IDM 跟驰模型<sup>[12, 13]</sup>，以及 MOBIL 换道模型<sup>[14]</sup>，并用粒子方法来模拟。在交通流模型中，智能驾驶员车辆跟驰模型(intelligent driver model, 下文称 IDM) 模型是一个空间连续，时间也连续的仿真方法，它既适用于高速公路，也适用于城市交通网络。是由 Treiber, Hennecke 和 Helbing<sup>[12, 15]</sup>在 2000 年假设加速度受到期望速度和期望跟驰距离的影响下提出的。

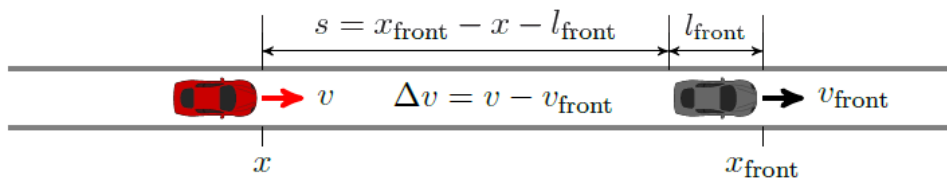


图 5-22

在 IDM 模型中, 车辆的状态由确定的时间, 位置, 速度, 及所在车道描述, 任何车的加速度只取决于它自身的速度以及离它最近的前方车辆。加速度是速度  $v$ , 间距  $S$  与前车的相对速度  $\Delta v$  (接近速度) 的连续函数:

$$\frac{dv}{dt} = a \left[ 1 - \left( \frac{v}{v_0} \right)^\delta - \left( \frac{S^*}{S} \right)^2 \right] \quad (1)$$

其中  $S^*$  为期望跟驰距离

$$S^* = S_0 + \left( vT + \frac{v\Delta v}{2\sqrt{ab}} \right) \quad (2)$$

上式 (1) 和 (2) 中的各参数定义如下:

$v_0$ : 车辆自由行驶的速度, 或可理解为车辆允许达到的最大速度。卡车具有较小的  $v_0$ 。

$T$ : 为期望 (最短) 车头时距, 车头时距为当前车辆以当前速度到达当前时该前方车辆位置所需要的时间。

$a$ : 最大舒适加速度。卡车具有较小的  $a$ 。

$b$ : 为最大舒适减速度。卡车具有较小的  $b$ 。

$S_0$ : 最短跟驰距离。

$\delta$ : 加速度指数。

下面简单的分析一下这个模型: 车辆的加速度可以分解为两个部分, 第一项是趋向自由行驶部分的加速度  $a \left[ 1 - (v/v_0)^\delta \right]$ ; 第二项是受前车影响部分的减速度  $-a \left[ 1 - (S^*/S)^2 \right]$ 。当实际的跟驰距离  $S$  小于期望跟驰距离  $S^*$  时, 第二项成为加速度的主要部分。

对于单车道 IDM 跟驰模型模拟, 本文没有做过多的参数分析, 主要以动画展示。图 5-23 所示是模拟结果, 其中圆环的半径为 120m, 车长为 6m, IDM 模型中的参数取值分别为:  $v_0 = 120\text{km/h} = 6\text{m/s}$ ,  $a = 0.5\text{m/s}^2$ ,  $b = 3\text{m/s}^2$ ,  $S_0 = 2\text{m}$ ,  $T = 1.5\text{s}$ 。仿真时间步长为 0.2s。环道上有一辆车在前 40s 内速度为 0, 从 40s 后开始行驶, 模拟前方的红绿灯由红灯变为了绿灯。

图 5-24 中小车的颜色表示的是速度。通过模拟还可以得到更多有趣的结果, 比如流量和时空轨迹等。另外要说明一下的是, 2009 年的交通环岛问题可以在此基础上加入几个进口和出口来进行模拟。

在上文中本节介绍了单车道 IDM 跟驰模型, 接下来本文将介绍双车道换道模型。这里, 只以 Minimizing Overall Braking decelerations Induced by Lane changes (下文简称为 MOBIL) [17, 14] 模型为例。如图 5-25 所示, 红车作为当前被考虑是否从右道换往左道的车, 标记为  $c$ , 作为换道前后旧的和新的后方车辆, 被分别标记为  $o$  和  $n$ 。在 MOBIL 模型中, 车辆要换道, 首先得满足安全条件: 换完车道后不能使后方车辆跟自己追尾。若红车  $c$  想换道, 则换道后必须满足

$$\bar{a}_n \geq -b_{\text{safe}}$$

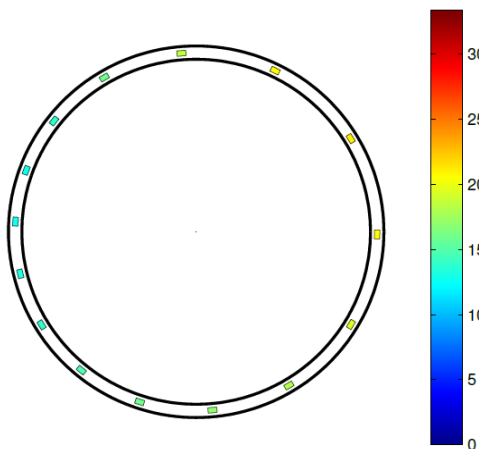


图 5-23

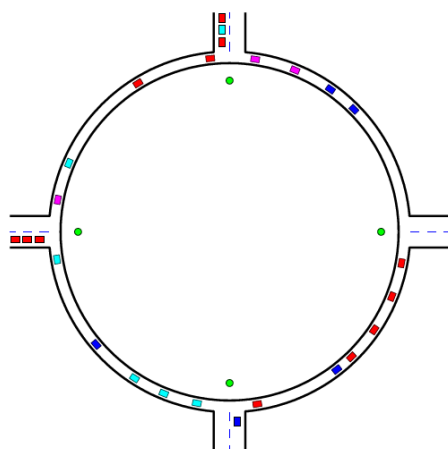


图 5-24

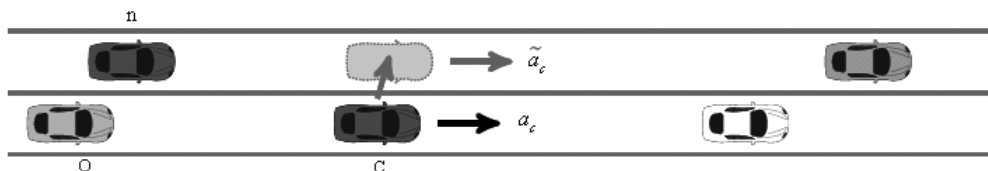


图 5-25

其中  $\bar{a}_n$  为红车  $c$  换道后灰车  $n$  的加速度（可能为负）， $b_{\text{safe}}$  为最大减速度。这表示换完道后不能让后方车辆减速超出最大减速度。在满足安全条件的情况下，对于对称换道（左右车道等同）和（非对称换道比如右行规则）分别有：

- 对称换道：换道的目的通常是让自己及自己局部的交通情况变得更好，特别是能让自己以一个更快的加速度或速度行驶。对于对称换道规则中，车辆换道还需满足

$$\bar{a}_c - a_c + p(\bar{a}_n - a_n + \bar{a}_o - a_o) > \Delta a_{\text{th}}$$

上式中的三小块分别是  $c$  车换道前后，引起的自身和紧随其后两个车辆加速度的变化值。其中  $p$  为一参数，如果  $p=1$ ，则上式为  $c$  车换道前后引起自身和紧随其后两个车辆加速度的变化值的总和。如果  $p=0$ ， $\Delta a_{\text{th}}=0$ ，则表示，如果换完车道后，只要当发现自己的加速度变大，就换道，这看起来不太礼貌，不顾及身后其他车辆的感受，因此  $p$  被称为礼貌系数（politeness factor）。

- 非对称换道（右行规则）：在对称换道的基础上，引入一个  $\Delta a_{\text{bias}}$  附加到  $\Delta a_{\text{th}}$  上即可。

从左往右换及从右往左换分别满足以下两式

$$\text{右换左: } \bar{a}_c - a_c + p(\bar{a}_o - a_o) > \Delta a_{\text{th}} - \Delta a_{\text{bias}}$$

左换右:  $\bar{a}_c - a_c + p(\bar{a}_o - a_o) > \Delta a_{th} - \Delta a_{bias}$

以上就是 MOBIL 模型的主要规则, 想要了解更多细节及讨论, 可以参考文献<sup>[17, 14, 18]</sup>。结合 IDM 跟驰模型和 MOBIL 换道模型, 可以模拟多车道下的交通流问题。这里本文展示两个例子。这两个例子的有关 IDM 模型的参数基本与图模拟所使用的一致。其他有关 MOBIL 参数分别为  $b_{safe} = 12$ ,  $\Delta a_{th} = 0.3$ ,  $p = 0.2$ 。图 5-26 所示的是由 IDM 跟驰模型和 MOBIL 换道模型实现的双车道右行规则的模拟。在右车道上有一辆速度较慢的卡车, 后方的车辆不得不经超车道以越卡车。图 5-27 所示的是由 IDM 跟驰模型和 MOBIL 换道模型实现的双车道对称规则的模拟。系统中共有三辆速度较慢的卡车, 后方的车辆不得左右交替换道以越过三辆卡车。

有关 IDE 和 MOBIL 模型的程序实现代码这里不再细述, 有兴趣的读者可以从 <http://www.traffic-simulation.de/><sup>[19]</sup>下载相关代码。

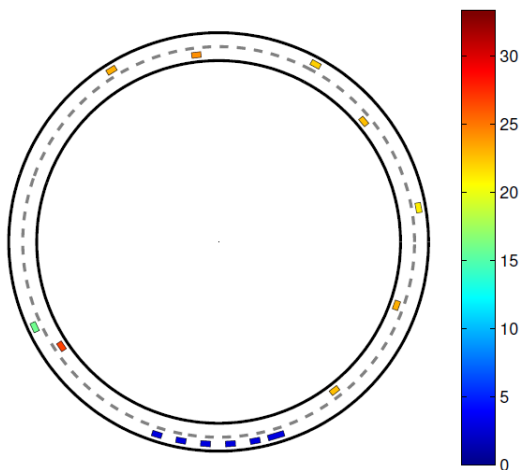


图 5-26

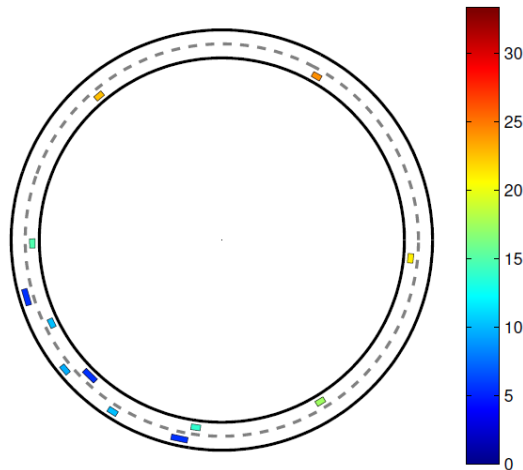


图 5-27

### 5.3.2 2014 MCM-A题特等奖论文

对于 2014 MCM-A 题, 本节给出一篇来自 Tufts University 的特等奖论文“Rules of the road: Using adaptations of Nagel-Schreckenberg Cellular Automaton Traffic Simulations to evaluate Passing Rules for Multi-Lane Freeways”, 并作简要评点。之所以选择这篇特等奖论文, 一方面是这篇论文模型简单, 并且不泛很多明显缺点, 却可能获得特等奖, 足见其有过人之处, 另一方面这篇特等奖论文是唯一一篇美国本土参赛队的特等奖, 有利于学习美国人的建模和写作思维。

#### 一、引言

由于高速公路上车辆的行驶速度很高, 出于最大限度地提高安全性和通行效率的目的,



运输监管机构通过交通规则限制多车道高速公路上任何时候司机的行为。比如有这样一条在汽车靠右行驶的国家普遍使用的交通规则：除非司机正试图超越前方的一辆车，否则必须在公路的最右车道上行驶。想要超车的车辆必需安全的换到左边，超过一个或多个要超过的车辆后，再回到原来所在的车道上。英国和澳大利亚等一些国家，使用了一个类似的规则：汽车靠路的左侧行驶。

本文尝试回答这样一个问题：“不超车就得靠右行驶”的规则是否是一个安全和有效的公路交通控制方法。其他三种潜在的规则分别称为：“无规则”，“禁止换道”和“居中行驶”；它们分别对应着允许车辆在各车道间自由换道行驶，随机分配车道并且禁止更换车道行驶，和可自由超车，但超过后得回到中间车道行驶。

在不同的交通条件下，包括较高和较低的车辆密度，较大和较小的限速，以及三种不同的事故发生率，以上三种规则都在相同的元胞自动机交通控制模型中实施，元胞自动机交通控制模型是基于 Nagel 和 Schreckenberg<sup>[20]</sup>提出的一种模型。对于每种规则和交通条件，事故数量和交通流量的数据是在 100 个单元区域，2 到 6 条车道上经过 250 个时间步长的统计得到的。这些模拟的结果以及模型的局限性和假设合理性的讨论如下文所述。

## 二、前人工作

### （一）交通问题的不同方法

交通流建模不是一个新问题，目前已经有很多模型被用来描述和模拟不同类型道路（从城市道路到高速公路）的交通流。维基百科上“交通流”<sup>[21]</sup>的页面单独列出了大量已被用于描述交通流的模型和参数，这些交通流的模型既有离散模型，也有连续模型。其中一种由 Doboszczak 和 Forstall<sup>[22]</sup>提出的模型用偏微分方程组来模拟基于平均密度的交通流量，而不是着眼于每个汽车。这种类型的模型用基本的方程组和假设来取代更复杂的数学描述。另一个模型由麻省理工学院的一个研究小组通过研究幽灵堵塞<sup>[23]</sup>现象建立，他们通过观察一组汽车并观测交通流量的行为，然后经验性的建立了规则来描述他们观察到的现象。

### （二）Nagel-Schreckenberg 模型

本中所应用的模型是源于 Kai Nagel 和 Michael Schreckenberg<sup>[20]</sup>在 1992 年发表的原创性工作，在他们的模型中，车辆被放置在一维的阵列中，阵列中的每一个元胞被车辆占据（非空），或者为空，每个非空元胞中的车辆都配有相应的速度，速度被限定在零到系统的最大限速之间。通常同一个元胞是不能同时被多个车辆占据，这个模型中每一个循环都同时执行四种基本的操作支配。这四种基本操作分别为：

#### 1. 加速

对于所有速度未达到最大值  $v_{\max}$ （道路的最大限速）的车辆，并且如果车辆前方有超



过  $v+1$  个空元胞, 则其速度增加一个单位。  $v \rightarrow v+1$

## 2. 减速以防止碰撞

如果一辆车的前方有  $d$  个空元胞, 且该车在下一时间步的速度超  $d$ , 则该车必需减速至  $v \rightarrow \min(d, v)$

## 3. 随机化

对于任何速度大于 0 的车辆, 其速度将以一定的概率  $p$  减小一个单位。  $v \rightarrow v-1$

## 4. 前进

对执行以上 1~3 步操作后, 根据当前速度  $v$  对所有车量指定一个新的位置  $x$ 。

$x \rightarrow x+v$

驾驶员只关心其目的地, 在这个层面上每一辆车的行为是独立的, 但是任何一辆给定的车辆的行为又取决于它对周围车辆的响应。这个模型假定任何一位驾驶员都期望以最快的速度行驶, 但同时驾驶员又希望避免事故。这个模型也引入了一个不完美的因素来控制驾驶员随机的降低他们所驾驶车辆的速度, 值得注意的是: 与车辆为避免与其前方车辆碰撞而强制减速类似, 这种随机的减速同样不会造成任何事故的发生。同样需要注意的是这个模型是为单车道交通系统设计的, 因此如果不作改进来考虑变道的可能, 该模型并不能用来解决本文所需要解决的问题。

# 三、模型建立

## (一) 简化性的假设

- 用少量车辆在一个预先定义好的较小的区域上的模拟, 可以被认为是一条完整高速公路的表示。
- 在每次模拟中, 没有任何车辆进入或离开高速公路, 也就是说, 对于任何一次模拟, 车辆的总数不变, 因此在整个模拟过程中, 所研究区域的车辆密度也保持不变。
- 所有车辆都是相同的, 并且它们在空间位置上相差车长的整数倍数。
- 可以通过修改事故发生的概率来考虑不同的天气状况(对交通的影响)。
- 驾驶员对即将发生事故的反应时间是因人而异的, 这里我们取随机值。
- 任何事故都发生在两辆车之间, 并且任何一个事故对其所在车道上的交通流的影响是相同的。发生事故的车辆在事故处理完后都将回到公路系统中(继续前进)。

第一个假设是必要的, 这是因为模拟大量的车辆在一个较长的距离上(的行为)计算的开销是非常昂贵的(主要是指费时间), 并且还需要一个复杂的、可能会显著依赖区域长度的体系来衡量交通流。这一假设在之前 Courage 等人<sup>[24]</sup>的论文中就被提出并被讨论过, 在他们的论文中, 车辆被看成是一小排区域上一起行驶的单元。第二和第三个假设是作为







本文基础的 Nagel-Schreckenberg 模型本身所固有的, 并且由于计算能力的限制, 这两个假设也是必要的。每次模拟都对不同元胞长度, 不同车辆密度及车辆总数进行计算, 对于这种模型, 这将会极度消耗资源。

第四个假设是由于本文的模型并没有明确的考虑不同的天气条件对高速公路上的事故发生率的影响, 尽管天气条件的确会改变与大量事故相关的行车条件。如果要求具体描述各种交通规则在大雨或暴雪天气中的表现, 我们将通过在某种较大的事故率条件下来测试其(规则)表现来解决这个问题。

第五个假设是用来允许车内驾驶员防止事故的发生。必须存在一个固定的反应时间值来防止危险的加速, 因此将在任一步长内, 对每一个驾驶员随机的指定一个反应时间。在 Benekohal 和 Treiterer 的 1988 年的论文里, 他们在 CARSIM 模型<sup>[25]</sup>中给出了一个使用随机反应时间的理由。虽然他们对指定随机反应时间的模型和参数远比我们的复杂, 但(我们和他们)使用随机反应时间的原因是一样的。

第六个即最后一个假设也是另一个能有效地降低每次模拟所需的时间的假设。虽然一辆车撞上已经和其他车辆发生碰撞的车是有可能的, 即三辆车连环碰撞。但这里我们假设不存在三辆车同时发生碰撞, 并且假设车辆不会与另一条车道上的车辆发生碰撞。这个假设的细节将在第 2.5 中讨论模型的局限性时具体讨论。

## (二) 模型

本文的模型应用 Nagel-Schreckenberg 模型中元胞的概念, 并在其基础上作添加以满足我们的需求, 设计这个模型的目的是用来比较不同驾驶习惯的效力。本文对不同行车条件的模拟, 是通过对一个  $n \times 3$  的矩阵进行一系列的操作, 其中  $n$  是模拟中车辆数。矩阵中的每一行表示一辆车, 其中第一列中的数字表示车所在位置的元胞标号, 第二列中的数字表示车辆所在的车道, 第三列中的数字表示车辆的当前速度, 如图 5-28 所示。

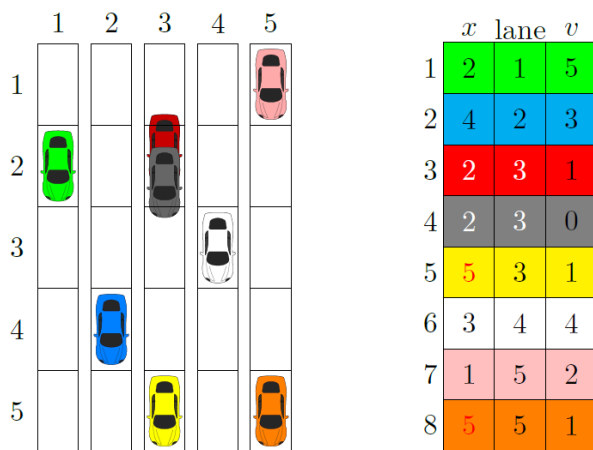


图 5-28

在 Nagel-Schreckenberg 系统基础上做的最大的改进是多车道系统的建模, 这对于比较



不同的换道策略是必要的。这使得多个车辆可能具有相同的元胞标号，如果它们的车道标号不一样。车辆运动的研究是在只含 100 个元胞的短区域上进行的，并包含一种机制来统计那些超出第 100 个元胞区域的车辆数，并使这些车辆返回到第 1 个元胞区域<sup>1</sup>。一个独立的系统来统计在一次模拟中的某个时间步内事故发生的数量，并且锁定车道上事故后方的车辆，强制其他车辆换道或者停止以等待事故现场的清理。

事故的发生是由于随机化操作导致的，这个随机化是由 Nagel-Schreckenberg 模型中的随机操作修改得到的。在原始的随机化操作中每一辆车有一个固定的随机减速的概率，前面已经提及，这种方式不会在系统中模拟出交通事故。在本文的模型中，车辆有一个固定的随机加速的概率，这导致它们可能会与它们前方的车辆追尾。此外，正如在上节中所述的那样，所有驾驶员会对潜在的事故做出随机的反应，当然这种反应可能会成功地防止事故的发生，也有可能失败。

本文的模型中，虽然对不同的交通规则有不同的一系列操作，但正如先前所述每一个操作都是对  $n$  行 3 列的矩阵进行修改，本文中元胞自动机的交通模拟中一般操作如下：

### 1. 加速

对于所有速度未达到最大值  $v_{\max}$ （道路的最大限速）的车辆，并且如果车辆前方有超过  $v+1$  个空元胞，则其速度增加一个单位。 $v \rightarrow v+1$ 。

### 2. 换道

对于允许变道的交通规则，如果一辆车的前方有  $d$  个空元胞，经过第一个操作后速度大于  $d$ ，且附近没有车辆阻碍其变道，那么这辆车就移到另一车道上，然后重复这个步骤，直到没有更好的车道可供移占。对于靠右行驶的规则系统中，在这一步操作中，车辆只能往左边车道上移。而对于无规则和居中行驶的系统，车辆可以不受限地向左或右车道移。

### 3. 减速以防止碰撞

如果一辆车的前方有  $d$  个空元胞，且该车在下一时间步的速度超  $d$ ，则该车必需减速至  $d \rightarrow \min(d, v)$ 。

### 4. 随机化

对于任何速度小于  $v_{\max}$  的车辆，其速度将以一定的概率  $p$  增加一个单位。 $v \rightarrow v+1$ 。

---

<sup>1</sup>笔者点评：原文作者将超出第 100 个元胞区域的车辆返回到第 1 个元胞区域，这无形在强制所有车辆都必需经过第 1 个元胞区域，笔者认为更合理的做法是使用周期性边界条件，使 100 个元胞构成一个环道。





## 5. 前进

对执行以上 1~3 步操作后, 根据当前速度  $v$  对所有车量指定一个新的位置  $x$ 。

$x \rightarrow x + v$ 。

## 6. 经过出口的车辆数及车辆的循环

如果某辆车在前进后, 其元胞标号大于 100, 而在前进前, 元胞标号是小于或等于 100 的, 通过这个系统的车辆数将增加 1, 同时如果有车道标号为 1 的元胞为空, 那这么辆车将被移到这个车道标号为 1 的元胞的位置上。如果不存在这样的车道, 这个车将在下一个时间步再回到起点。Throughput  $\rightarrow$  Throughput+1, Cell  $\rightarrow$  Cell+1。

## 7. 检查并处理事故

如果发现两辆车在同一个条车道且具有相同的元胞标号, 事故的计数就填加 1, 并且这两辆车的速度设为不同的两个负值。对于具有负速度的车辆, 跳过除了“加速”操作外的所有其他操作, 这将使涉事车辆在一段固定的时间内被冻结在事故发生的位置, 以模拟清理事故现场并使涉事车辆重回到道路所需的时间。Accidents  $\rightarrow$  Accidents+1,  $v \rightarrow v - 1$  或  $-5$ 。

## 8. 回到最初指定的车道

本操作步只在靠右行驶和居中行驶的规则系统中执行。对于每一辆车, 都会检查这辆车是否在其初始所指定的车道上, 1 表示靠右行驶规则, 2 或 3 则表示居中行驶规则, 如果某辆车不在其指定的车道上, 并且在朝着其指定车道方向上的侧边车道上没有车辆, 那么该车将向着其指定车道方向移动一个车道。这个操作将被重复直到不能再靠近指定车道为止。Lane  $\rightarrow$  Lane $\pm$ 1

这个模型是在 MatLab 环境下, 通过执行上述功能的一系列脚本建立的, 每一个时间步至少执行一个脚本, 在不同的交通规则系统中, 执行不同组合的脚本, 这些脚本可以以模块的方式插入到一个函数中, 以方便建立一个能运行多次并对四种交通规则系统中的每种的结果数据求平均的函数。

# 四、结果

## (一) 参数及术语

所有展示的数据都是本文对每种规则的模型模拟 10 次的平均结果。每一次模拟都是在 100 个元胞的区域上, 250 个时间步长内。驾驶员避免事故所需的反应时间是固定的, 使得每个驾驶员在某次潜在事故中成功避免的可能为 30%。所有其他的参数都被认为是可以变化的, 测试所有可能的条件组合的情形。

(1) 最大速度,  $v_{\max}$  (公路的限速)。

快——对大限速情况, 设置  $v_{\max}$  等于 6 元胞/时间步。

慢——对小限速情况, 设置  $v_{\max}$  等于 2 元胞/时间步。

(2) 引起事故的随机加速概率。

智能-速度随机改变的概率被人为的设置为 0, 这意味着发生事故的概率也为 0。

冲动-任何一辆给定的车的在任何给定的时间步中随机加速的概率被设置为 0.3。

正常-任何一辆给定的车的在任何给定的时间步中随机加速的概率被设置为 0.01。

(3) 交通密度。

高-在模拟区域放置有大量的车辆, 这里我们定义为 100 辆。

低-在模拟区域放置有少量的车辆, 这里我们定义为 30 辆。

对于 4 种交通规则系统中的任一种, 5 条车道中的任一条的“平均流量”的计算都是通过统计模拟期间通过模拟区域的车辆总数(模拟十次求平均), 然后这个总数除以模拟的总时间步数, 即 250 步。“平均事故量”为一次模拟中事故发生的总数, 也是 10 次模拟的平均值。

## (二) 图

接下来的几页中展示的图汇总了分别从 4 种交通规则的模拟中获得的数据, 每一种交通规则的模拟都涵盖了不同的限速, 事故发生率, 交通密度, 车道数情况。这些图是按照交通规则分别展示的, 每一张图都附有该规则模拟条件(这里的条件指限速, 事故发生率, 交通密度, 车道数)的简要说明。

## (三) 靠右行驶

靠右行驶规则表示右车道用于行驶, 左车道用于超车。所有的车辆都是从最右边的车道上开始行驶的, 就像所有的车都刚刚进入高速公路一样。然后这些车为了超过慢车而分散开来, 但是一有机会这些车就会试图回到最右边的车道上。靠右行驶规则的平均流量和事故量如图 5-29 和图 5-30 所示。

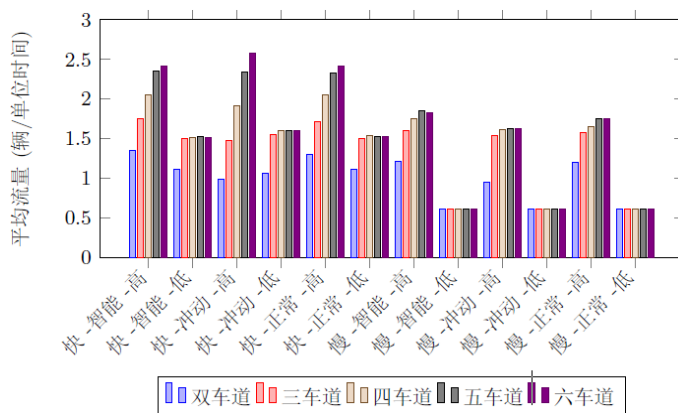


图 5-29

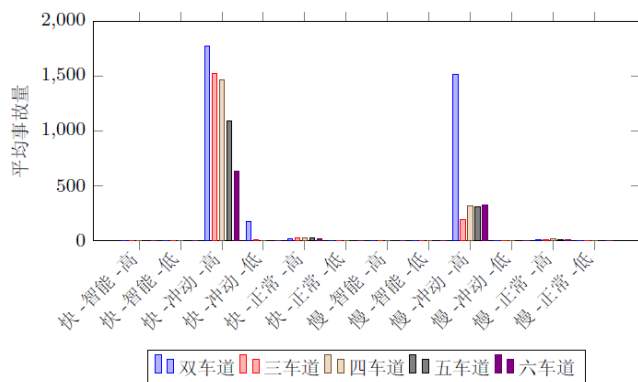


图 5-30

#### (四) 无规则

无规则是指不将向左换道和向右换道区分优先次序的规则。在这种规则的模拟中，车辆初始被设置在随机的车道上，并且只有当它们要超车时才会换到别的车道上。无规则行驶的平均流量和平均事故量如图 5-31 和图 5-32 所示。

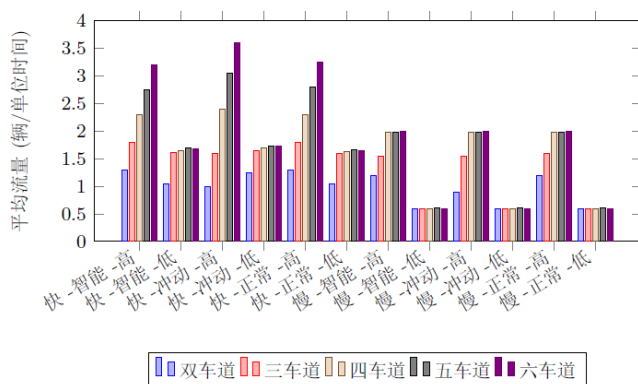


图 5-31

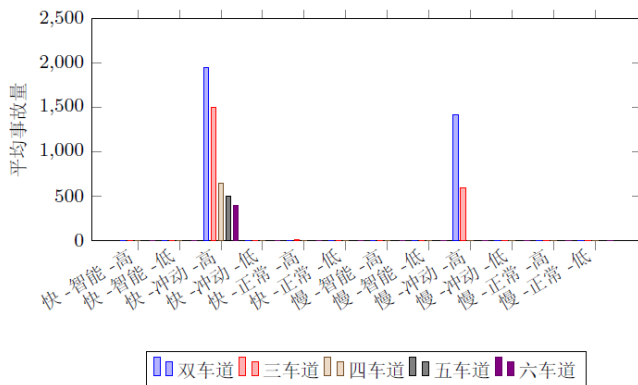


图 5-32

（五）禁止换道

禁止换道是指完全不允许任何车道改变其车道的规则。这种规则下不存在超车，所有车的初始车道都是随机生成的，就如同它们进入高速公路中的一条随机的车道，并且它们必须留在这一条车道上直到它们决定离开高速公路。所有换道的操作步骤都被忽略。禁止换道规则的平均流量和平均事故流量如图 5-33 和图 5-34 所示。

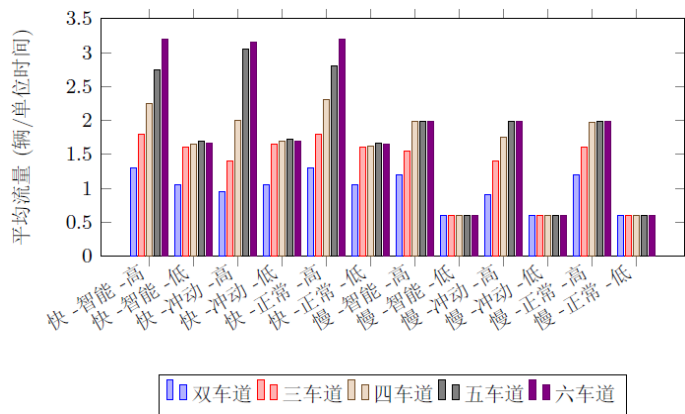


图 5-33

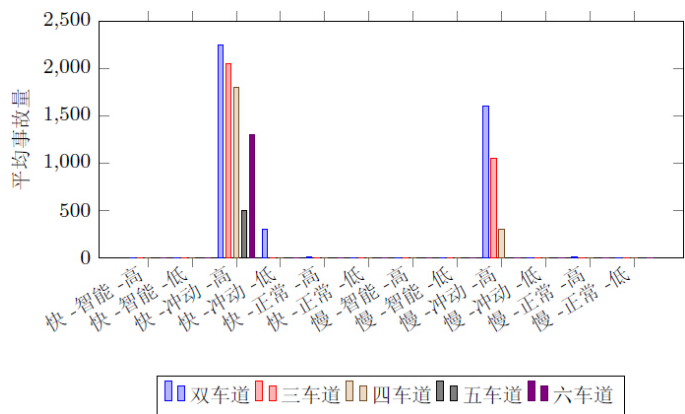


图 5-34

（六）居中行驶

居中行驶规则中，所有车道初始都被放置在高速公路的最中间车道上，然后它们散开以超过别的车辆。超车的机制类似于“无规则”，车辆可以从左车道或者右车道超车，但超车后，这些车辆始终会尝试寻找高速公路上中间车道上的空位（换回中间车道）。居中行驶规则的平均流量和平均事故量如图 5-35 和图 5-36 所示。

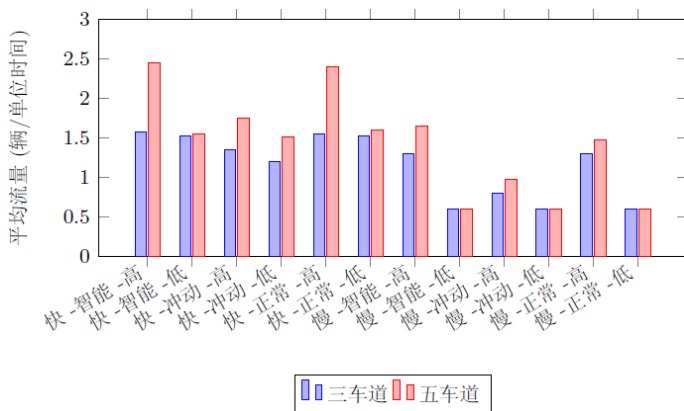


图 5-35

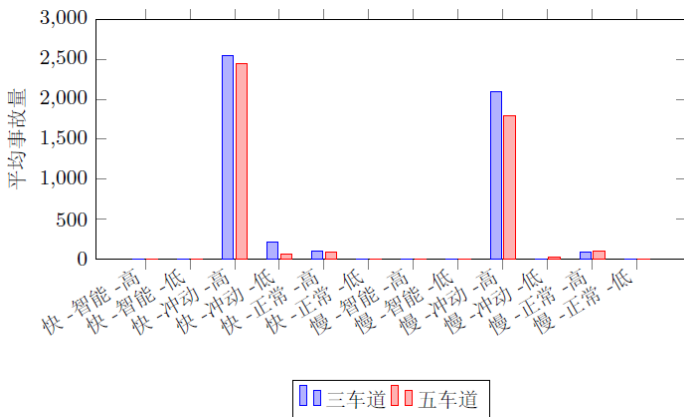


图 5-36

### (七) 不同交通规则的比较

靠右行驶规则是美国标准的交通规则，与之相比较的规则有以下三种不同的交通规则。

- 无规则。

无规则模拟允许驾驶员从公路左右两边的任一条车道上超车，前且可以留在任意车道上。与其他所有规则相比，这个规则给出了最大交通流量，不过在车道数较少时，相比于靠右行驶规则（RHR）有那么一点点不安全。无规则具有最大的流量并不完全出人意料，这因为无规则并不像禁止换道规则，它允许所有车辆做任何可以使自己加速的事，并且允许换道以越过一个交通事故。无规则模型中的车辆不会在任何循环的最后尝试回到最初的车道上，这允许它们占用任何可用的车道以获得更快的速度，这也是该规则在车道数较多的模拟中能够获得较大的交通流量的原因。在车道数较少时，相比于靠右行驶规则（RHR）会多引起那么一点点事故，这可能是因为车辆频繁地在其他车辆前面换来换去车道导致的。不过，当车道数较多时，相比于靠右行驶规则(RHR)，无规则会表现的更安全，在靠右行驶规则中，车辆强制自己尽可能的靠右行驶，因此增加了事故发生的可能。

- 禁止换道。

禁止换道规则是所有规则中最简单的，这个规则在初始时生成一个固定数量的车辆随机分布在各条车道上，在整个模拟中车辆都将保持在自己的车道上（行驶）。这实质是同时运行了多条一维的 NS 模型。这个模型给出了交通流量第二大的结果，但却也是最危险的规则。

模拟中车辆的运动是分解为离散操作步骤完成的，其中有一个步骤是随机加速。当一个事故发生成禁止换道的模型中，事故后方的车辆不能换道以越过事故，而其他所有规则的模型却可以。这个事实导致的结果是：当随机加速的概率和车辆密度同时较大时，单个交通事故将导致其后大量的车辆卡住并在五个时间步长内保持速度为 0。由于较大的事故发生率，某一辆车终于经过随机加速后，与前方车辆发生碰撞，引起了事故，最终导致了连锁反应。这一点可以从居中行驶中的第二幅图中看出，“Accident-Prone Heavy Traffic”的模拟结果显示出事故障量非常大。在同一张图中，随机加速概率为 0 或者 0.01 的模拟表现出较少或没有事故。

有趣的是，国家生物技术信息中心的 Alexander 在其博客<sup>[26]</sup>里用自己改进的 NS 模型的模拟结果显示：“呆在自己现在的车道”的规则系统比车辆往速度快的车道上换能获得更大的平均速度。他只讨论了这两种方式，“换到快速的车道”的规则系统并不直接与“无规则”系统类似，他的模型是在一个真正的环道上实施的，但他并没有尝试计算事故的发生率。但是，他的结论提供了一个有趣的对“禁止换道”规则性能的重新审视：相比于换到更快车道的规则系统，“禁止换道”规则系统总体上为交通系统保持了一个相对较大交通流量。尽管他的模型有很多不同，但是他的模型仍然支持了由本节的数据得到的结论，相对于某些其他的换道系统，一个从来不换车道的交通系统会使所有车辆都较快的行驶。

禁止换道是本节所提供的最不切实际的规则，这是因为它根本不允许驾驶员换道，甚至在发生交通事故时，但这一规则仍然是有用的，它可以作为一个参照来评价其他模型的性能，估计其他换道规则对系统的交通流量的总体效果。

- 居中行驶。

在所有模型中，居中行驶规则表现为流量最小，同时也表现为最大的事故率。由于偶数车道的高速公路并没有单个的最中间的车道，这个规则的数据是从三条和五条车道的高速公路的模拟中收集到的，收集到的数据强有力证明：这个规则引起较大的事故发生率，并且相比于所研究的其他交通规则，该规则下通过整个区域的车辆总数最小。居中行驶和靠右行驶规则非常相似，从三条车道增加至五条车道并不会像无规则和禁止换道规则那样大幅度地降低事故发生率。对于被研究的四种交通规则，这个被认为是目前为止最差的一个。

## 五、讨论

### （一）模型的局限性

由于这里的假设和算法建立的方式，在模型中很多因素无法正确地考虑。故这里的模型是分步执行的，这意味着模拟中所有的车都是独立的，车辆并不同时移动。因而展示的







模型完全是基于NS 模型建立的,而NS 模型的等时间步长的不确定也被带到了我们的模型设计中。本文模型的一个缺点就是算法中的每一操作步骤都是逐个执行的(例如车辆是一个一个的移到左边,然后一个一个的检查是否安全)<sup>2</sup>。只有在加速操作过程中,当一辆车与前方车辆碰撞时才会发生事故,这意味着将所有的交通事故都归为追尾事故,且车辆在换道时不会发生任何事故。

事实上并不是这样,事故可以发生在任何方向上,但是这些都源自所有事故都是随机发生的假设。

模型中计算事故数量的方式依赖于随机数的产生。车辆的加速或减速取决于产生的随机数是否小于模型中某一指定的参数,如果随机生成的驾驶员反应时间不是足够的短就会发生事故。我们构造了一种完美的驾驶方式(即智能驾车)用于模拟,因此我们假设驾驶员完全掌握周围的交通信息,近而他们随机加速的概率为 0。对于“正常”情况的模拟,我们设置事故发生的概率低至 0.01。这假设驾驶员会控制他们的车辆并且会踩刹,很少或者完全不会在驾车时分心。在本文冲动驾车模式的模拟中,随机加速的概率被设置为 0.3,在任何情况下,(这么大的概率都表明)驾驶员明显很少控制他们的车辆。在任何情况下,我们都没有更改为避免事故所需的反应时间。不同的天气状况需要不同的反应时间,以及不同的随机加速概率。路面有冰的情况要比大风时候更危险。我们并不研究有关天气对交通影响的更多细节,而用冲动模式的模拟来涵盖所有这些情形。

在我们的模型中,当矩阵  $\text{Car\_Matrix}$  中出现两辆车具有相同车道和相同位置值时表示发生了事故。这将引起同一条车道上事故后方的车辆停滞,以模拟整个交通的减慢。然而,在一些模拟中,某些确定的交通事故会连锁反应地引起更多的交通事故。这是由本文添加的随机加速引起的,当车辆被连续地排列在各自的车道上,如果事故后方存在多辆等待事故现场清理的车,并且发生了随机加速(这种事在 *accident prone* 模拟中很常见),随之而来的就是另一场事故并继续阻碍车道上的交通。在禁止换道的模拟中,当发生事故后车辆不能离开自己的车道,这确实是一个相当大的问题,这导致了更多的事故出现在了禁止换道的模拟中。

车辆及其行为是由矩阵  $\text{Car\_Matrix}$  来模拟的,借助 *MatLab* 编程,用矩阵来模拟的一个主要限制是:矩阵中车辆数不能超过表示高速公路车道元胞的行数。因此,我们不能正确地模拟完全阻塞的网格,以及真实的交通车量密度。这个限制在多条车道的高速公路的模拟结果中显得更为显著。对于不同车道数的模拟中,车辆数保持为常数,格子里的车辆密度随着我们增加车道数而减小,即便是在被认为“交通堵塞”的情况下,这也意味着交通越来越畅通,由于车辆之间的相互作用越来越少使得通行无阻,因此事故的发生量减小,并且车流量增加。尽管如此,这个模型仍然是可用的,因为随着循环的增加,由于要前进,车辆间将会越来越近,因此在元胞组成的网格的顶部会出现车辆聚集,这形成了一种伪堵

---

<sup>2</sup>笔者点评:由于算法中的每一操作步骤都是逐个执行的,计算机循环是有方向的,因此换道也是优先次序,这也是本文的一大缺点,不过这完全可以用很简单的方法解决:对要换道的车辆进行随机排序,再按随机顺序依次换道。



塞。另外，由于在所有模型中车辆密度的都是以相同的速率减少，所有模拟中得到的所有数据都经过了同样条件的变化。

模拟过程中，没有新的车辆进入或离开高速公路系统。此外，车辆初始进入高速公路时要么全部分布在同一条车道上，要么根据模拟需要随机分布于每条车道上。然后所有的车都待在高速公路系统中，并且只向前移动。实际上高速公路上不停地有车进入或离开，这意味着车辆有理由越过车道，而不是强制地服从交通规则。如果汽车经过出口，那么它将更可能发生事故，相比于本文的任何模拟，车辆会更大程度地分散在元胞网格上。

## （二）模型的总结

本文模型中所有的交通规则都可以容易地过渡到靠左行车的交通系统中。在那些靠左行驶的国家，车辆的驾驶员座位在另一边，因此司机的能见度是一样的，并且我们的规则同样适用，因为我们的模型中完全只用了整数来表示不同的车道，位置和速度，而且本质上我们的算法是对称的（即左右没差别）。因此我们可以通过重新反向给矩阵 `Car_Matrix` 中的车道编号来模拟车辆靠左行驶，用 `n:1` 取代 `1:n`，而模型中的算法仍然有效。

## （三）灵敏性分析

在任何一个特定的模拟中，本文的模型似乎都对车道数非常灵敏，这时由于在高速公路区域上生成车辆的功能不能生成超过元胞数量的车数，即便在多条车道的模拟中。这个限制由MatLab的`randperm` 函数（`randperm` 是MatLab 函数，功能是随机打乱一个数字序列）造成的，并且无法在本模型范围内避免<sup>3</sup>。这也导致了在车道数较多时，两种不指定初始车道的规则（无规则和禁止换道）表示为非常安全，虽然在车道数较少时并不是太安全。这也同样导致了对于六条车道的高速公路平均每6个元胞才出现一辆车，因此对于那些能够将车辆分散在所有车道上的规则，车辆密度变得非常小，而对于那些强制车辆回到某个指定车道上的规则，车辆密度则保持相对较高。

模型中另一个相对较为灵敏的因素是控制模拟中事故发生率的概率。虽然这个参数只测试了三种值，但仍可以观察出事故数量与控制事故发生率的概率呈非线性关系。模拟中一旦有少量的事故发生，它们似乎还会导致更多的事故发生。这种现象在车辆密度较大，限速较高及随机加速概率较大的所有模拟中都会导致极高的事故率，远高于基于其他条件值导致的事故量的期望。其部分原因可能是：为了模拟最坏交通行驶条件的情况，这些模拟中我们设置了非常大的随机加速概率。但是这不能完全解释我们所看到的数据结果，这里面一定还有部分原因是事故的连锁反应。

---

<sup>3</sup> 笔者点评：原文作者把模型的限制部分归结为MatLab中的函数`randperm`，笔者不以为然。`randperm` 是MatLab 中生成随机排列序列的函数，`randperm`函数有两种用法：

- 第一种：如`randperm(6)`则可生成1到6的随机排列，结果可能为[2 4 5 6 1 3]；
- 第二种：如`randperm(6,3)`则是从1~6个数中随机选三个，结果可能为[4 2 5]。

原文作者一定是`randperm(100,n)`来生成 $n$ 个车的初始位置。一条车道上格子的个数为100，因此 $n$ 必然是小于100。因此对于多条车道，系统中车量数最多也只能是100。这就是作者说的灵敏性的原因所在。





#### (四) 结论

基于呈现在我们面前的所有数据，可以得到这样一个结论：现在使用的靠右行驶的交通规则是最安全的一种驾驶方式。我们的数据还向我们显示了：允许向两个方向换道，或者完全不允许换道将带来更大的交通流量，但比靠右行驶危险。由于我们优先考虑安全驾驶并到达他们的目的地，因此我们认为无规则和禁止换道以增加事故的数量为代价，增加交通流量是不值得的。

### 5.3.3 程序实现

在本书的第4章中已经介绍了单车道 NS 元胞自动机模型的 MatLab 实现方法，在本章节的 1.2 中，介绍了双环形车道交通流的元胞自动机 NS 模型的 MatLab 实现方法。这里给出包含事故的单车道 NS 模型的仿真程序。程序和不含事故的单车道 NS 模型的仿真程序非常类似，主要是增加了一个函数 `accident` 来产生和计算事故率。以下是主程序：

```
1 function Nacdnts = nsacdnt(rho, p, L, tmax, pacdnt)
2 % 输入: rho - 密度; p- 随机减速概率; L- 环形车道长; tmax - 模拟总时间步数;
3 % pacdnt - 随机加速概率.
4 % rho = 0.25; p = 0.25; L = 100; tmax = 200; pacdnt = 0.001;
5 % 输出: Nacdnts - 事故量
6
7 vmax = 5; % 车辆最大行驶速度
8 ncar = round(L*rho); % 根据密度计算系统中车辆数
9
10 x = sort(randperm(L, ncar));
11 v = vmax * ones(1,ncar); % 速度向量: 初始化所有车辆的速度为 vmax
12 isacdnt = zeros(size(x));
13
14 Nacdnts = 0;
15 for t = 1:tmax
16     v = min(v+1, vmax); % 加速规则
17
18     gaps = gaplength(x,L);
19     v = min(v, gaps -1); % 防止碰撞
20
21     vdrops = ( rand(1,ncar)<p & v>0);
22     v = v-vdrops; % 随机减速
23
24     vadds = ( rand(1,ncar)<pacdnt & v>=0);
25     v = min(v+vadds, vmax); % 随机加度
26
27     x(v>0) = x(v>0) + v(v>0); % 位置更新
28     x(x>L) = x(x>L) - L; % 周期边界
29
```



```

30     [isacdnt , v, nacdnt] = accident(isacdnt , v, x);
31     Nacdnts = Nacdnts + nacdnt;
32 end

```

以上主程序 `nsacdnt` 的第 24 和第 25 是随机加速规则，这与 NS 规则中随机减速实现方式类似。第 30 行调用 `accident` 函数，函数 `accident` 是用来构造事故的，在随机加速后，可以出现两辆车在同车道相同位置，`accident` 函数将这样的车辆的速度设为负。而第 30 行位置的更新只更新速度大于零的车辆，因此必需经过数次加速直到使事故车辆速度大于零才前进，以模拟处理事故所需的时间。下面给出 `accident` 函数的代码：

```

1  function [isacdnt , v, nacdnt] = accident(isacdnt , v, x)
2  % 输入: isacdnt - 表示车辆是否处于事故状态的变量
3  % 输出: nacdnt - 处理事故的车辆数
4  for xi = unique(x);
5      tot = find(x==xi); % 所有发生事故的车
6      old = find(x==xi&isacdnt ==1); % 之前未处理完的事故车辆
7      new = find(x==xi&isacdnt ==0); % 新发生事故的车辆
8      if length(tot) >=2 % 发现两辆以上的车处同一位置
9          if isempty(old) && ~ isempty(new) % 全部是新事故车辆
10             vmin = -1;
11             v(new) = vmin - 5*[0: length(new) -1];
12             else % 有之前未处理完的事故车辆,也可能有新事故车辆
13                 vmin = min(-1,min(v(old)));
14                 v(new) = vmin - 5*[1: length(new)];
15             end
16             isacdnt(tot) = 1; % 标记事故车辆
17         end
18     end
19     isacdnt(v>0) = 0; % 取消处理完事故车辆的标记
20     nacdnt = sum(isacdnt); % 计算当然处理事故中的车辆数

```

函数 `accident` 中 `isacdnt` 是一个向量，该向量中每个元素对应着一辆车，若该向量中某元素为 1，则表示其对应的车辆处理事故。函数 `accident` 中第 5 行中 `tot` 表示处于 `xi` 位置的所有车辆，第 6 行中 `old` 表示处于是 `xi` 位置之前未处理完的事故车辆，第 7 行中 `new` 则表示新发生事故的车辆。第 8 行表如果处于同一位置 `xi` 的车辆数大于 2，则表示有车辆处于事故状态。有车辆处于事故状态又分为两种情况：

- 第 9~11 行，表示如果处于事故状态车辆都是本时间步新发生事故的车辆，则将处于事故状态车辆速度分别设为 -1 和 -6。
- 第 12~15 行，表示如果处于事故状态车辆有之前未处理完的事故车辆，也可能有新事故车辆，则将处于事故状态车辆速度分别设为 `vmin`，`vmin-5`，`vmin-10`...





### 5.3.4 论文点评

下面就这篇特等奖论文笔者作简要点评，这篇特等奖论文的模型描述得非常清楚，虽然没有任何插图，但读者仍然能清楚地了解到模型的全部。这篇特等奖论文的模型虽然有不少局限性（这些局限性在前面已作详细讨论），但模型的可操作性非常强。

这篇特等奖论文的最大创新点是事故的构造，NS 模型并不容易直接计算事故的发生率，比如清华大学的特等奖论文就采用了间接的做法：用急刹车的频率来间接反应事故的发生率，在防止碰撞规则中，统计车辆速度降低 2 以上的频率。清华大学的特等奖论文认为车辆在急刹车的状况下，显然更容易造成一个现实中的事故。而本篇特等奖论文将 NS 规则中的随机减速改为了随机加速，自然引入了交通事故。

在写作上，这篇特等奖论文对各规则的流量和事故率分别做了分析和对比，讨论部分也非常精彩，很客观的评述了模型的优缺点。并分析了造成缺点的原因。

## 5.4 2014 MCM-B

笔者是 2014 年美国大学生数学建模竞赛特等奖（Outstanding & Frank Gioradno Award）获得者熊风。熊风所在参赛队的论文是今年 B 题唯一入选官方杂志 *umap journal* 的论文，本节将由熊风细述他们队对 2014 年 B 题的见解以及特等奖论文的具体思路。

### 5.4.1 问题综述

MCM 的题目不长，允许我全文摘录于此。

PROBLEM B: College Coaching Legends

Sports Illustrated, a magazine for sports enthusiasts, is looking for the “best all time college coach” male or female for the previous century. Build a mathematical model to choose the best college coach or coaches (past or present) from among either male or female coaches in such sports as college hockey or field hockey, football, baseball or softball, basketball, or soccer. Does it make a difference which time line horizon that you use in your analysis, i.e., does coaching in 1913 differs from coaching in 2013? Clearly articulate your metrics for assessment. Discuss how your model can be applied in general across both genders and all possible sports. Present your model's top 5 coaches in each of 3 different sports.

In addition to the MCM format and requirements, prepare a 1-2 page article for Sports Illustrated that explains your results and includes a non-technical explanation of your mathematical model that sports fans will understand.

拿到题目之后，我建议大家把题目原文反复认真地读上几遍，分析一下题目的要求和要点。我归纳一下我们队当时整理的内容：

Sport Illustrate 是 Time Warner 媒体集团的一个美国体育媒体杂志。该杂志现想要挑选



出近 100 年来男女“优秀的大学教练”。最好教练的挑选范围不论性别，不论领域，比如美国足球、冰球、橄榄球、英式足球、篮球以及曲棍球教练均可作为候选人。

我们需解决的四个主要问题：

- 选择我们的度量方法以及建立相应数学模型；
- 建立评价教练表现的评价系统；
- 在讨论时间因素、性别因素以及领域因素的情况下，我们模型的运用情况；
- 分析各种参量的印象度，并讨论我们模型可以拓展运用的范围。

对数学建模竞赛比较有经验的同学，应该能比较迅速地意识到这是一个评价类型的问题。关于数模竞赛中的评价问题，可以用的数学模型还是不少的，模糊评价、层次分析法、TOPSIS 法、秩和比法，等等。

同样，评价类型的赛题在以前的国赛和美赛中也曾出现过不少，较近的就有 2010 年数学建模的国赛题“上海世博会的影响力定量评估”。经过了这些讨论，我们在比赛的第一天下午明确了一个结论：这个题目容易做，但是比较难做出亮点。

数学建模中的有的题目比较难，但是只要能做出来，一般很容易出亮点，比如一些需要用到比较高端算法的问题。而有些题目比较简单，甚至求解它们都存在一些固定的套路，但比较难做出亮点。我们觉得 2014MCM 的 B 题就属于后一种。我们可以猜到，绝大部分参赛队会用到的模型就那么几种吧，而且肯定能做出一个结果；那我们又该如何脱颖而出呢？且看下文分析。

## 5.4.2 分析与建模及求解

### 1. 比赛过程中的思考

在比赛第一天的晚上，我们已经确定做这题的整体策略：我们队无法别出心裁地去弄出一个新的模型，因此想做好就一定要把那些熟知的模型做得尽可能深入和完善。初步考虑，我们会使用层次分析法，这是一个很经典、很常见的评价模型，我在之前的比赛中使用过多次；但是，层次分析法的缺点也是显而易见的，中间有很大一部分是专家打分。而这部分我们不可能有权威的数据，只能人工来打分，所以这部分是相当主观的。为了弥补层次分析法过于主观的缺点，我们打算再建一个模糊综合评价模型。我个人在模拟赛中使用过这个模型，并且也保留有核心部分的代码。所以说，这两个模型我都比较熟悉，而且手里都有现成的求解代码，这在比赛中大大节省了时间。在模型方面既然无法有创新之处，我们自然就想把用来评价的指标体系做得更复杂，精确一些。

而所有的指标都是要搜集具体的数据，因此这题找数据是一个比较关键的地方。如果对于涉及的体育运动不是太了解，想到的数据可能只有教练的生涯胜场、负场、拿过的冠军。仅仅靠这些数据是很难建立一个完整的指标体系，模型也会显得特别干涸。

所以在比赛的第一天晚上和第二天上午，我们花了大量时间了解体育背景和查找数据。以篮球为例，对于一个 NCAA 教练来说，什么数据能够衡量他的成绩和地位呢？如果以最

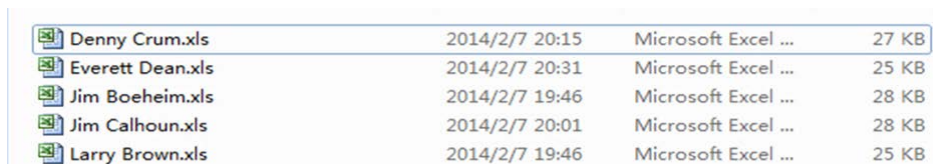






粗糙的处理方法，或许我们想到的只是胜场和冠军数。但仔细一想，这显然是不够的。胜场也应该考虑含金量，考虑比赛的重要程度；而 NCAA 的成绩也不能只用冠军数来衡量，分区冠军，十六强，八强，四强，亚军、总冠军这些成绩都应该有具体的量化。所以，要确定如何找数据，就要先对这项赛事有充分的了解。只有在真正了解这个比赛的情况下，才能把握哪些数据是关键，哪些数据含金量高、占的权重大。

在这个过程中，我们一方面去各个网站查阅资料，了解规则；一方面去 NCAA 贴吧向那些 NCAA 老球迷请教问题。弄清了很多数据（比如 SRS, SOS）的意义，也能够把握 sweet sixteen、final four 等战绩的难度和含金量了。最后，为了让数据尽可能地详细，我们查找了每个教练整个职业生涯每个赛季的详细数据：这个赛季的胜场，负场，SRS 值，SOS 值，最后的成绩……如图 5-37 所示。



Denny Crum.xls	2014/2/7 20:15	Microsoft Excel ...	27 KB
Everett Dean.xls	2014/2/7 20:31	Microsoft Excel ...	25 KB
Jim Boeheim.xls	2014/2/7 19:46	Microsoft Excel ...	28 KB
Jim Calhoun.xls	2014/2/7 20:01	Microsoft Excel ...	28 KB
Larry Brown.xls	2014/2/7 19:46	Microsoft Excel ...	25 KB

图 5-37

我们把每个教练的生涯数据都单独做成了一个 Excel 表，仅针对 NCAA 这项赛事，我们就整理了几十个 Excel 表格。那么为什么是找这几十个教练的信息呢？喜欢篮球的朋友应该都知道，美国篮球有个名人堂，在 NCAA、NBA 有卓越成绩的球员和教练都会入选名人堂。既然是评选 NCAA 历史前五的教练，那候选名单肯定在篮球名人堂中。所以我们在找数据的时候，只用搜集入选名人堂的 NCAA 教练的数据。这里要注意一下，我们也处理了一些特殊情况。比如 Tom Izzo 这个教练，他的生涯战绩和荣誉都相当耀眼，完全可以进入名人堂；但我们在名人堂名单里却没有找到他。去问了资深球迷，才知道一般教练是要退休一段时间之后才会进名人堂；有些教练虽然取得了很辉煌的成绩，但是可能暂时未进入名人堂，等到以后入选。虽然这部分教练的数量比较少，我们还是把他们的数据添加进候选数据集了。

由于我们找数据力求详细，所以这个过程费了我们三个人很长时间。但是最后的数据也比较齐全，以篮球为例，每个教练的数据向量都在十维以上。但这个时间花得是值得的，也是有必要的；正是因为详细的数据，我们才能把模型的指标体系量化部分做得完善，对后文的建模和求解都有很大的帮助。

## 2. 具体的指标量化和建模

这一部分内容很大部分是我们获奖论文的中文翻译，如果想详细了解，建议直接阅读我们的获奖论文原文（在校苑数模论坛上可以找到）。

### （1）指标量化。

对于球员的评价，已经有了比较成熟的体系，那就是五项能力分析，包含力量、速度、技巧、防守和进攻五个方面。类似的我们可以从下面五个方面确立教练的评价指标：生涯



纪录、竞技含金量、季后赛表现、荣誉以及对此项运动的贡献。下面将具体阐释这五个方面。

- 生涯纪录：教练在整个执教生涯中带队的成绩毫无疑问是评价这个教练最重要的因素之一。而在众多纪录指标中，胜场、负场最能直接说明问题。

胜场总数 $a$ 可以按下式计算：

$$a = \sum_i a_i$$

- $a_i$ 代表第 $i$ 年的胜场数。

负场综述 $b$ 可以按照下式计算：

$$b = \sum_i b_i$$

- $b_i$ 代表第 $i$ 年的负场数。

- 竞技含金量：整个赛程的难易程度可以代表这个赛程的含金量。胜弱队并不一定能说明问题，而在战胜强队的同时能保持较高胜率这样的战绩才是令人信服的。同时如果在战胜对手的同时，保持较高分差，同样能说明这个队伍的统治力。出于这个考虑我们选取了下面两个指标，来表征这个含金量。
- ◆ Simple Rating System (SRS)：SRS 系统主要统计手段是计算一个队伍的平均胜分。我们将教练执教生涯中每个赛季的 SRS 做一个统计平均，作为第一个评价指标。

$R$  表征平均 SRS，那么  $R$  可以按照下式计算：

$$R = \frac{\sum_i SRS_i}{t}$$

- $SRS_i$  代表第 $i$ 年的 SRS 值， $t$  代表总年数。

- ◆ Strength of Schedule (SOS)：赛季强度 (SOS) 主要统计手段是计算每一场比赛前两队的实力差距，来表征本场比赛的强度。

$O$  表征平均 SOS，那么  $O$  可以按照下式计算：

$$O = \frac{\sum_i SOS_i}{t}$$

- $SOS_i$  代表第 $i$ 年的 SOS 值， $t$  代表总年数。

- 季后赛表现：通常来说季前赛只能表征赛区内队伍的竞技成绩，但是和其他赛区的对于交手机会并不多（因不同球类赛制而异），因此季后赛成绩应该在评价一个教练的成绩过程中占一个很大的比重。对于美国高中篮球季后赛，可以引入下面这些指标：

- ◆ 第一轮：季后赛一轮出局的队伍。
- ◆ 第二轮：季后赛二轮出局的队伍。
- ◆ 十六强：季后赛进入十六强的队伍。
- ◆ 四强：季后赛四强队伍。
- ◆ 亚军：在决赛中失败的队伍。
- ◆ 冠军：在决赛中获胜的队伍。



为了对本方面的指标进行量化，我们统计某教练他的队伍季后赛每一种成绩的次数  $n_k$ 。我们用二进制变量  $m_{ki}$  代表第  $i$  年季后赛是否获得第  $k$  种成绩（一轮  $k = 1$ ，冠军  $k = 7$ ），那么  $n_k$  可以按照下式计算：

$$n_k = \sum_i m_{ki}$$

- 荣誉：教练的荣誉值可以用很多种奖项来表征，篮球名人堂和大学篮球名人堂也可以列为重要荣誉。为了量化荣誉值，我们某教练获得各种荣誉的次数，例如 [Naismith College Coach of the Year, Basketball Times National Coach of the Year]，等等。考虑到不同的奖项可能有不同的含金量，我们确立了一个奖项权重 ( $h_i$ )。  $\mathcal{H}$  表征总的荣誉值，那么  $\mathcal{H}$  可以按照下式计算

$$\mathcal{H} = \sum_i h_i$$

- 对该项体育的贡献：这个概念是宽泛的，我们主要选取的以下五个方面来量化这个贡献
- 培养的明星球员数目
- 执教生涯长度
- 重要的战术革新
- 在其他国际比赛中的表现：比如在世锦赛、奥运会中的表现。
- 知名度：以人名为关键字的 Google 词条数

每个方面的分数用  $c_i$  表示，取值 0 到 2：0 代表碌碌无为，1 代表一般，2 代表卓越。这样我们可以得到贡献值总分  $\mathcal{C}$ （满分 10 分）

$$\mathcal{C} = \sum_i c_i$$

图 5-38 中集中体现了上面阐述的指标。



图 5-38

## （2）层次分析模型（AHP）

下面，我们展示了包含准则层以及子准则层的递阶层次结构。



Goal	Criteria	Alternatives
The influence of coach	Historical Record	Wins
		Losses
	Game Gold Content	SRS
		SOS
	Playoff Performance	First Round
		...
		Champion
	Honors	Different Awards
		Hall of Fame
	Contribution to sports	Star Player
		Coaching Age
		Tactical Innovation
		International Games
		Popularity

The three hierarchy structure of our model

获得评价因子的权重

- 构造判断矩阵

我们使用配对比较法以及 1~9 标度的方式去构造判定矩阵  $A = (a_{ij})$ 。

$$a_{ik} * a_{kj} = a_{ij}$$

其中 $a_{ij}$ 由 1~9 标准进行设定。

- 计算特征值以及特征向量

$\lambda_{max}$ 是矩阵  $A$  的最大特征值，并且相应的矩阵向量是 $u = (u_1, u_2, u_3, \dots u_n)^T$ 。我们用下面公式对  $u$  进行归一化。

$$x_i = \frac{u_i}{\sum_{i=0}^n u_j}$$

- 一致性检验

计算一致性指标  $CI$

$$CI = \frac{\lambda_{max} - n}{n - 1}$$

其中  $n$  代表矩阵的维度。

计算一致性比例  $CR$

$$CR = \frac{CI}{RI}$$

鉴于我们已经确认了评价系统中各个指标的权重大小，我们接下来进行量化教练们的重要程度。

$CW_i$ 代表第  $i$  个标准层指标的权重大小， $AW_j$ 代表第  $j$  个子准则层指标的权重大小。

评价分数衡量公式 $Y_1$



$$Y_1 = \sum_{i=1}^5 CW_i \times \sum_{j=1}^{mi} AW_j \times F_j$$

这其实是层次分析法的标准步骤，上述的模型只是把我们的指标体系代入了层次分析法中而已。

### (3) 模糊综合评价模型 (FSE)

模糊数学理论自 1965 年开始就得到了发展和运用 (Zadeh, 1965)。主要是为了对一些比较宽泛的指标进行量化描述。

在上面部分我们已经指定了五个方面考虑的指标，包括生涯纪录、竞技含金量、季后赛表现、荣誉以及对该运动的贡献。在使用模糊综合分析法之前，我们先用下面的计算准则得到分数向量  $\{X_1, X_2, X_3, X_4, X_5\}$ 。这个工作的目的主要是将第三部分的每一类小指标汇聚起来，形成该方面的总分，方便进行后续的模糊数学的处理。

- 生涯纪录的计算准则：

$a$  代表胜场数， $b$  代表负场数， $\lambda_{\text{win}}$  代表单胜场的权重， $\lambda_{\text{lose}}$  代表单负场的权重，那么本方面的评价分数可以按照下式计算

$$X_1 = \lambda_{\text{win}} a - \lambda_{\text{lose}} b$$

- 竞技含金量的计算准则：

$R$  代表平均 SRS， $O$  代表平均 SOS， $O_{\text{max}}$  代表 SOS 最大值。那么本方面的评价分数可以按照下式计算：

$$X_2 = R \left( 1 + \frac{O}{O_{\text{max}}} \right)$$

如果一个教练拥有较高的 SRS，那么这方面的分数会得到明显提升，用来代表教练的统治力。同时 SOS 会带来难度系数加成，赛程越难，加成系数越高。

- 季后赛表现的计算准则：

$n_k$  代表取得某类型的季后赛成绩的次数，那么本方面成绩可以按照下式计算

$$X_3 = \sum_{k=1}^7 2^k n_k$$

由于参加各轮次比赛的队伍数目按照 2 的指数次幂下降，我们设定成绩的权重按照 2 的指数次幂上升。

- 荣誉的计算准则：

在第三部分我们已经计算了  $\mathcal{H}$ ，这里我们之间借鉴过来作为本方面的分数。

$$X_4 = H$$

- 对该运动的贡献的计算准则：

在第三部分我们计算了一个本方面的总分  $C$ ，这里我们同样借鉴过来，作为本方面的分数。

$$X_5 = C$$

总的来说，我们各方面的计算准则如下

$$\begin{cases} X_1 = \lambda_{\text{win}} a - \lambda_{\text{lose}} b \\ X_2 = R \left( 1 + \frac{O}{O_{\text{max}}} \right) \\ X_3 = \sum_{k=1}^7 2^k n_k \\ X_4 = H \\ X_5 = C \end{cases}$$

确定隶属度函数

模糊集的定义依赖于隶属度函数，隶属度函数的主要作用是把不同的值映射到 0~1 区间内，其大小可以表征某个值隶属于该模糊集的程度。

$X_{ij}$  表征第  $i$  个教练的第  $j$  个方面的评分  $X_j$ ， $X_{j(\max)}$  代表所有教练  $X_j$  的最大值。

这里我们采用归一化的隶属度函数

$$\mu_j(X_{ij}) = \frac{X_{ij}}{X_{j(\max)}}$$

对于每个  $X_{ij}$  计算对应的  $\mu_j(X_{ij})$  后，我们就可以得到模糊矩阵  $X_f$  ( $N$  代表教练总数)

$$X_f = \begin{bmatrix} \mu_1(X_{11}) & \dots & \mu_1(X_{15}) \\ \dots & \mu_j(X_{ij}) & \dots \\ \mu_N(X_{N1}) & \dots & \mu_N(X_{N5}) \end{bmatrix}$$

用熵权取值法确定权重

熵权取值法主要有以下五个步骤：

- 对于  $X_{ij}$  (归一化后) 计算特征比重  $p_{ij}$

$$p_{ij} = \frac{X_{f(i,j)}}{\sum_{i=1}^N X_{f(i,j)}}$$

- 第  $j$  项指标的熵值  $e_j$  的计算

$$e_j = -\frac{1}{\ln(N)} \sum_{i=1}^N p_{ij} \ln(p_{ij})$$

- 指标  $j$  的差异系数  $g_j$  的计算

$$g_j = 1 - e_j$$

- 确定指标的权重系数  $w_j$

$$w_j = \frac{g_j}{\sum_{j=1}^5 g_j}$$

- 综合分数对于每一个教练的综合评分  $Y$

$$Y_2 = W^* X_f$$

隶属度函数的确定其实还可以有别的选择，这里我们选择了形式比较简单的一种。而熵权取值法也是一个比较固定的套路，在这里我们没有什么创新的地方。

### 3. 模型综合

把两个有明显缺陷的模型摆出来，并不代表解决了问题。为了结合两个模型的优点，





我们决定建立一个综合模型。而综合的最基本方法就是线性加权，关于权重的确定，在这里我们引入了一个新的指标：对模型的评分。

为了对两个独立模型的准确度进行评价，我们定义了平均偏移距离 $D$ 。

我们从几家权威媒体（例如 ESPN，露天报道，雅虎体育和体育新闻）的评选中收集到了 NCAA 篮球教练的前十排名。然后把我们的结果同这些排行榜的结果作比较，用平均偏移距离来反映我们结果同排行榜之间的差异。

这里我们用一阶闵可夫斯基距离来表征前十名的平均偏移距离。

$$D = \frac{1}{10n} \sum_{i=1}^n \sum_{j=1}^{10} |j - r_j|$$

其中 $n$ 是排行榜的数目， $j$ 是第 $i$ 个排行榜的排名， $r_j$ 是第 $j$ 个教练在我们结果中的排名。所以 $|j - r_j|$ 表示的是我们和媒体结果的差异， $D$ 表示平均偏移距离。如果我们的结果和每个媒体评选的结果都一样，那么 $D$ 等于0。

$D_\alpha$ 是前5名的平均偏移距离

$$D_\alpha = \frac{1}{5n} \sum_{i=1}^n \sum_{j=1}^5 |j - r_j|$$

$D_\beta$ 是第6名到第10名的平均偏移距离。

$$D_\beta = \frac{1}{5n} \sum_{i=1}^n \sum_{j=6}^{10} |j - r_j|$$

显然平均偏移距离小的模型获得的评分应该更高。所以我们可以定义命中得分为

$$g = \frac{900}{9+D} \quad (0 < g < 100)$$

其中 $D=0$ 的时候， $g=100$ 。意味着如果平均偏移距离为0，模型可以得到满分。

层次分析法是一种主观的方法，它主要依赖于人为打分。相对地，模糊综合评价是一种纯客观的方法，它全部基于数据。为了更好地考虑主观和客观的因素，我们采用线性加权法

$$\begin{cases} W_1 + W_2 = 1 \\ Y = W_1 Y_1 + W_2 Y_2 \end{cases}$$

$Y_1$ 是层次分析法的评分， $Y_2$ 是模糊综合评价法的评分，它们的范围都是0~100。

为了更好地确定权重，我们考虑平均偏移距离。因为更小的平均偏移距离代表着更准确的结果，我们把较高的权重分配给较小平均偏移距离的模型。我们可以得到

$$\begin{cases} W_1 = \frac{D_2}{D_1 + D_2} \\ W_2 = \frac{D_1}{D_1 + D_2} \end{cases}$$

我们的最终评分被定义为

$$Y = W_1 Y_1 + W_2 Y_2$$

以上就是我们的最终模型，其实思路就是简单的线性加权。只不过我们的权重并非随

意设定，而是在这里定义了一些新指标，建立了一个小模型来确定的。这也是数学建模比赛比较灵活的地方，很多东西没有一个标准的答案，允许你自由发挥，脑洞大开：你可以定义新的指标，新的评价方式，只要你言之有理，能自圆其说即可。

#### 4. 求解

层次分析法和模糊综合评价模型（主要是后面的熵权取值法）都能够找到现成的代码，而事先我也有收集。在赛前收集常用算法的代码，我觉得是比较必要的工作。但是注意，代码一定要保证自己调试过，使用过；如果平时收集的代码到比赛的时候发现不会用，或者有 BUG，会损失很多不必要的时间。

在模型求解这一块，我们队基本没花什么时间。我只需要写一些简单的数据处理程序，并且针对题目的具体要求小幅度改动下赛前收集的代码即可。

下面给出核心部分的代码（MatLab）

##### (1) AHPSolver

```
function AHPSolver
HierarchyNum=input('Hierarchy Num of Ahp\n');
NumOfElem=input('Number of element in every Hierarchy\n');
weightKtoTop=0;
for i=1:HierarchyNum-1
    sprintf('plese input the DecMatrix of  %dth Hierarchy',i)
    k=0;
    while true
        flag=input('Do you want input DecMatrix in this Hierarchy? 1 or 0\n');
        if flag==1
            k=k+1;
            sprintf('plese input the %d row %d th: DecMatrix',i,k)
            sprintf('the position of 支配元素 in %d th 层 ',i)
            DecMatrix(k).DecEle=input('\n');
            sprintf('the position of 被支配元素 in %d 层 ',i+1)
            DecMatrix(k).SupEle=input('\n');
            DecMatrix(k).Matrix=input('DecMatirx \n');

            [DecMatrix(k).Weight,DecMatrix(k).LamdaMax,DecMatrix(k).CR]=AHPWeightVector(DecMatrix(k).Matrix);
            disp('the weight of this DecMatrix:');
            DecMatrix(k).Weight
            disp('the DecMatrix.LamdaMax:');
            DecMatrix(k).LamdaMax
            disp('DecMatrix.CR:');
            DecMatrix(k).CR
        else
            disp('this DecMatrix in this Hierarchy Input over');
            break;
        end
    end
end
```





```

end

if i==1;
    weightKtoTop=DecMatrix(1).Weight;
    CR=DecMatrix(1).CR;
else
    for j=1:k
        CR=CR+weightKtoTop(j)*DecMatrix(j).CR;
    end
    U=zeros( NumOfElem(i+1),NumOfElem(i));
    for j=1:k
        U(DecMatrix(j).SupEle,j)=DecMatrix(j).Weight;
    end
    weightKtoTop=U*weightKtoTop;
end
sprintf('the weight of  %d  Hierarchy element is',i+1);
weightKtoTop
end
end

```

## (2) AHP

```

% 五个大指标对总目标的影响
AA = [ 5 1 7 5 5];
[n,m] = size(AA);
A = zeros(m);
for i=1:m
    for j=1:m
        A(i,j)= AA(i)/AA(j);
    end
end
A(1,3) = 3/5;
A(3,5) = 5/3;
[WeightVector0,lamdaMax0,CR0]=AHPWeightVector(A)

% 胜场、负场的影响
BB = [3 -1];
[n,m] = size(BB);
B = zeros(m);
for i=1:m
    for j=1:m
        B(i,j)= BB(i)/BB(j);
    end
end
B(1,2) = 3;
B(2,1) = 1/3;
[WeightVector1,lamdaMax1,CR1]=AHPWeightVector(B)

% SRS、SOS 的影响
CC = [3 1];
[n,m] = size(CC);
C = zeros(m);
C(1,2) = 3;
C(2,1) = 1/3;
[WeightVector2,lamdaMax2,CR2]=AHPWeightVector(C)

```

```

for i=1:m
    for j=1:m
        C(i,j)= CC(i)/CC(j);
    end
end
[WeightVector2,lamdaMax2,CR2]=AHPWeightVector(C)

% first round-- champion 对战绩的影响
DD = [1 2 4 8 16 32 64];
[n,m] = size(DD);
D = zeros(m);
for i=1:m
    for j=1:m
        D(i,j)= DD(i)/DD(j);
    end
end
[WeightVector3,lamdaMax3,CR3] = AHPWeightVector(D)

% 个人荣誉

% 额外加成

x = [
% 具体的数据
];
[m,n] = size(x);
for i=1:n
    mmin = min(x(:,i));
    mmax = max(x(:,i));
    r(:,i) = (x(:,i))/(mmax);
end
newX = zeros(m,5);
newX(:,1) = r(:,1:2)*WeightVector1;
newX(:,2) = r(:,3:4)*WeightVector2;
newX(:,3) = r(:,5:11)*WeightVector3;
newX(:,4) = r(:,12);
newX(:,5) = r(:,13);
ans = newX*WeightVector0

```

### (3) 模糊综合评价

```

clear;
clc;
x=[
%% 具体的数据
];
%隶属函数的确定
[m,n] = size(x);
for i=1:n
    mmin = min(x(:,i));

```



```

    mmax = max(x(:,i));
    r(:,i) = (x(:,i)-mmin)/(mmax-mmin);
end
for j=1:n
    for i=1:m
        if(r(i,j)<=0.1)
            r(i,j) = r(i,j) + 0.1;
        end
        p(i,j)=r(i,j)/sum(r(:,j));%熵权法赋值
    end
end

for j=1:n
    e(j)=-1/log(5)*sum(p(:,j).*log(p(:,j)));
end
for j=1:n
    g(j)=1-e(j);
end
for j=1:n
    w(j)=g(j)/sum(g);
end
w(2) = w(2)-0.04;
w(4) = w(4)+0.01;
w(5) = w(5)+0.03;
w(1) = w(1)-0.02;
w(3) = w(3)+0.02;
B=w*r';

```

注意在上述的代码中，层次分析模型的求解，只需要把输入数据填入（2）中的  $X$  矩阵；模糊综合评价模型的求解，只需要把输入数据填入（3）中的  $X$  矩阵。在我们比赛的程序里都是有具体的数据的，但是由于数据量较大，在这里没帖出来。所以如果你直接使用上面的代码，注意给  $X$  矩阵赋值。 $X$  矩阵相当于输入数据的接口。

### 5.4.3 论文点评

看完我们的分析、建模和求解部分，想必大家都不大了解：这篇论文为什么能拿 Outstanding 呢？虽然搜集的数据比较齐全，建立的指标体系比较完备，但是关键的建模和求解部分都没有太大亮点。模型也就是很常见的层次分析法和模糊综合评级模型，最后再来了一个简单的线性加权得到一个综合模型而已。这种思路其实很一般，即使细节部分处理得不错，那最多给一个一等奖，也不至于拿到特等奖吧？

大家注意，我们的论文在获得特等奖的同时还获得了一个单项奖 Frank Gioradno Award。关于这个奖的官方说明是 This award goes to a paper that demonstrates true excellence in the execution of the modeling process。也就是说这个奖是授予模型执行过程中真正卓越的论文。确实，我们的论文创意方面其实不足，但我们很努力地把简单常用的模型做出深度，分析到位。我觉得我们论文最大的亮点不在于模型建立和求解，而在于模型的结果分析。这是





全文最深入，最出彩的部分。下面我给大家做一个具体分析。

建完模型，得到结果，对于此题是显然不够的。因为题目中已经明确要求，需要我们讨论时间因素、性别因素、领域因素对模型的影响。所以，我们需要对模型和结果做一些拓展和分析。

不妨看看我们论文的中文版目录，如图 5-39 所示。

IV. 排名的两种模型.....	7
4.1 Model I: 层次分析模型(AHP).....	7
4.1.1 递阶层次结构.....	7
4.1.2 获得评价因子的权重.....	8
4.1.3 结果及分析.....	9
4.2 Model II: 模糊综合评价模型 (FSE).....	10
4.2.1 五方面成绩量化:.....	10
4.2.2 确定隶属度函数.....	11
4.2.3 用熵权取值法确定权重.....	11
4.2.4 结果与分析.....	12
V. 模型综合.....	12
5.1 对独立模型的评分.....	12
5.2 综合模型.....	13
5.3 结果和分析.....	14
VI. 模型扩展.....	15
6.1 性别无影响.....	15
6.2 时间因素的影响.....	15
6.2.1 为什么时间因素会产生影响?.....	15
6.2.2 时间因素如何产生影响?.....	16
6.2.3 时间因素带来的变化趋势.....	19
6.3 模型也适用于其他运动.....	19
VII. 深入分析.....	21
7.1 对模糊综合评价法的敏感性分析.....	21
7.1.1 改变隶属度函数.....	21
7.1.2 改变计算规则.....	23
7.2 对综合模型权重的敏感性分析.....	24
7.3 拓展: 评选最好的总统.....	25

模型建立和求解部分一共用了 7 页篇幅（第 7~第 15 页），但我们的模型拓展和深入分析部分竟然用了 10 页篇幅（第 15~第 25 页）！我们组拿到这个题目的时候就大概确定了我们论文的优势不在前面的建模求解部分，而在后面的分析部分。因此这部分的篇幅比建模求解本身还要长。

首先，我们论证了时间因素确实会造成影响。根据 NCAA 历史的转折点，我们把过去一个世纪分为六个时期，赋以不同的时间权重。在引入时间权重后，我们总结得到了排名的变化趋势。为了验证模型的通用性，我们将模型应用到美国大学女子篮球的教练排名中，发现性别因素不会造成影响。同时模型也使用于其他运动，对美国大学橄榄球和曲棍球也进行了分析，得出了结果。

然后，我们还对模型进行了多方面的敏感性分析：在模糊综合评价模型中，依次改变隶属度函数和计算规则，观察结果的变化。接着，改变综合模型中两个模型的权重，利用敏感性分析结果证明了综合模型确实比单个的模型表现更好。最后，我们还将模型应用于美国总统评价，选出了美国历史上得分最高的三个总统：林肯、华盛顿、罗斯福。一方面，这使文章内容更充实，更有深度；另一方面，我们这也算是投其所好。根据我们对美国的



了解，美国群众似乎比较喜欢调侃议论他们的总统，所以我们觉得把模型应用到评选美国总统中，有可能让阅卷者眼前一亮。

我们这样的论文能够拿特等奖，能够作为 B 题唯一的一篇论文入选 UMAP 杂志，或许能给大家一些启示。由于 Outstanding 获奖比例实在太低，很多人或许神话了 O 奖得主的实力，也神话了 O 奖论文本身。大家都觉得 O 奖论文必有重大创新和不同凡响之处，O 奖得主肯定是积累非常深厚的学神。其实我们能拿 O 奖，说明了一个实力 M 奖水平的队伍如果能在细节上面做的更好一些，结果分析做得更深入一些，再加上些许的运气，也是有希望拿到 O 奖的。

所以我想纠正很多选手的一个误区。我见过不少数模选手，比赛太追求奇技淫巧了。一些明明可以用简单、常用的方法就可以很好解决的问题，他们却偏偏要用一些比较生僻，比较高端大气上档次的方法。他们以为如果论文里没有一些特别高端的模型和算法，就基本和大奖无缘了。实际上我觉得不是这样的。同样的模型和算法，你能把它做得更详细，更深入，细节做得更好，也足以显现自己的功力，也有希望拿到 O 奖。

所以我个人的经验，想获得比较好的奖，论文肯定需要有亮点的。但亮点不仅是指新颖、先进、高逼格的模型和算法，亮点也可以是对问题完善的考虑、对细节的把握和追求，对结果各个角度与维度的深入分析。甚至精美的表格和图形同样也能是大大加分的亮点。一篇数模论文出彩的办法有很多，而我觉得结果分析是能稳定能做出亮点的一部分。因为很多时候，你即使没有思路，也至少可以调整下模型的几个关键参数，做一下敏感分析，画出一些横向、纵向对比的图形，这部分的工作做得好就能让论文的深度上一个档次。因此我觉得结果分析是一个很重要的部分；大家不要觉得完成了建模、求解的工作就万事大吉了，事实上结果分析也需要花大力气去做，需要花同等的精力去对待。

以上便是我对 2014 年 B 题以及我们队论文的分析，希望能对大家有所启示。愿每一位数学建模参赛选手享受比赛的过程，取得满意的成绩。

## 5.5 2013 MCM-C

这是 2013 年的 ICM 题，也就是 C 题。题目要求非常明确，直接要求建立一个网络模型来解决此问题。

相比之前 5 年所有的美赛题，都没有这样要求使用特定模型的，从这点看，此题其实相当固定，并没有太多发挥空间。所以，对于此题，各位选手一定要对网络模型有一定程度的了解。在此推荐一本适合初学者了解网络分析模型的书，《社会网络分析法》，重庆大学出版社译，书中较为细致地解释了网络分析法的各个方面。

但是，对于这题，单单用网络分析模型是不够的，因为网络模型一般重点在分析网络中的组成成分及其性质，如点、线、密度、位置、结构、成分以及维度，一般都是静态的。而这题很明显是动态的，所以，选手们也必须对其他各种模型有一定程度的了解，知道其应用的方式，以及如何将其与网络进行结合。

接下来，笔者将根据自己当时做题的思路，来对这道题进行分析、建模、求解和点评。



### 5.5.1 问题综述

#### 1. 问题原文

要让各位读者对这道题有全名的了解，首先，必须清楚题目所给的信息和要求。在这里，为方便阅读，我们已经将原文进行了翻译，但是，笔者建议真心为求知求提高的读者们，去找来原文仔细地看看。

#### 背景

当今社会对于发展模型和使用模型来预测我们星球上的生物科学及环境卫生非常感兴趣。许多科学研究表明，地球的环境及生物系统压力越来越大，但是目前只有极少数可以描述全球的模型可以用来测试这些问题。由联合国支持的《千年生态系统评估综合报告》发现，近三分之二的地球维持生命的生态系统——包括干净的水、纯净的空气和稳定的气候等，因为被不可持续的使用，正在退化。大部分损坏得归咎于人类：不断飙升的食品、新鲜水、燃料和木材需求也引起了戏剧性的环境变化，还有森林砍伐、空气、土地和水的污染。尽管相当多的研究对本地栖息地和区域因素进行了研究，但是当前的模型并没有充分告知决策者，他们狭隘的政策可能会如何影响整个地球的健康。许多模型忽略了复杂的全球因素而无法确定潜在的政策长远影响。而科学家们意识到在种种环境和生物系中，复杂的人际关系和交叉效果影响了地球的生物圈，当前模型经常忽略了这些关系或限制了系统之间的连接。系统的复杂性体现在多个交互作用，反馈循环，紧急行为或即将到来的状态改变或临界点。最近 22 国写的自然文章知名科学家题为“走近地球的生物圈”的状态转变列出了很多相关的问题，如对科学模型的需求，如何与预测潜在的行星卫生系统状态的变化相联系。本文提供了两个具体的量化建模的挑战，要求利用它们更好的预测模型：

(1) 通过全球性的模型，此模型包括了地球上的相互关联系统的复杂性和对当地条件的影响，以此来提高生物科学预测，反之亦然。

(2) 确定可能会产生不健康的全局状态变化的因素，并显示如何使用有效的生态系统管理，以防止或限制这些即将发生的状态变化。

我们研究的问题是：我们是否可以通过使用本地的或地球健康状况的组成元素来建立全球模型，从而预测潜在状态的变化，并帮助决策者根据他们的潜在影响，对地球上的生态保护设计出有效的政策。虽然许多警告现象出现，但是没有人知道地球是否真的接近全球的临界点，或者这种极端的状态是否是不可避免的。

这篇自然文章和其他许多人指出，有许多重要的元素对地球的生态系统有影响（例如，本地因素的影响，全球性的影响，多维因素和关系，不同时间和空间尺度）。也有许多其他因素可以被包括在一个预测模型中——人口、资源和栖息地压力、栖息地改造、能源消耗、气候变化、土地利用模式、污染、生物多样性、海洋化学、大气化学、政治模式如社会动荡和经济的不稳定。古生物学家研究和模拟了生态系统在以前的灾难性状态下发生的变化行为和反应，因此基于历史的定性和定量信息可以为未来的预测模型提供背景。然而，应该指出的是，在我们目前的生物圈状况下，人类的影响显着增加。





## 要求

您是交叉学科建模竞赛（ICM）的一名成员，ICM 将很快举办一个题为“网络和地球的健康”的比赛，要求您对此进行建模和分析。要求您的团队做到以下几点：

（1）建立一个关于地球健康某些方面因素的动态全球网络模型（您要促进这些措施的实施）通过识别此条件下的局部元素（网络节点）和适当连接（网络链接），以追踪关系和属性效果。由于这些效果的动态性质是非常重要的，这个网络模型必须包含一个动态过程，还可以允许该模型预测该卫生措施的未来状态。例如，节点可以是国家、大陆、海洋、栖息地或任何这些或其他元素的组合，共同构成全球性的模型。随着时间的推移，您的链接可以代表节点或环境的影响，或传播或繁殖的物理元素（如污染）。您的卫生措施可能是地球条件内的任何元素，包括人口、生物、环境、社会、政治、物理或化学条件。要确保定义了您的模型中的所有元素，并解释模型中网络结构、节点实体和链路属性的科学依据。确定一种方法来设置参数，并解释如果有足够的数据如何测试您的模型、什么样的数据可以用来验证模型或验证模型的有效性？（注：如果您没有必要的数据来确定参数或进行验证，不要否决你的模型。出题者意识到，在这个阶段，良好的创意思想和理论与验证基于数据的模型一样重要）。请确保您的模型包括了人为因素，并解释人类行为和政府政策怎么影响模型的结果。

（2）运行您的模型看看它是如何预测未来的地球健康状况。您可能需要估计参数，您通常会根据确定的数据来做决定。（请记住，这仅仅是测试和了解您的模型的元素，而不是使用它来进行预测或决策。）您的模型会产生何种因素？它能预测在地球的条件下状态的变化或临界点吗？它能否进行改变当地条件的全球性后果的警告吗？它是否可以告诉决策者在重要政策上如何决策？在您的措施和网络属性中您是否考虑了人为因素？

（3）使用网络建模的强大原因之一是能够分析网络的结构。网络性质是否可以帮助确定关键节点或在模型中的关系？如果是的话，进行这样的分析。您的模型如果缺少环节或改变关系，敏感性如何？您的模型是否使用反馈回路或需要考虑不确定性？数据收集问题是什么？请问您的模型是否反应了政府的各项政策，并可能从而有助于规划？

（4）写一个 20 页的报告（汇总表不计算在这 20 页里）来说明你的模型及其价值，一定要有模型的优点和缺点分析。如果这份报告是合适的和富有洞察力的，并且可用来进行行星健康问题建模，出题者将来可能会使用你的报告作为主题，并且会邀请你参加即将举行的研讨会。在接下来的网络建模工作中，祝好！”

## 2. 问题分析

美赛中，与生态环境相关的主题，最近几年出现地非常多，不得不承认的确出题者对于这些有着很高的重视度。

地球环境和生物健康在物种生存与进化、人类生活质量和社会繁荣方面扮演着至关重要的角色。并且环境变化和人类社会的相互作用已经持续了几千年，有着错综复杂的悠久历史。但是，这种相互作用于上世纪发生了根本上的变化。现在，人类活动所造成的影响无处不在，严重威胁着我们赖以生存的生态系统，使得地球向着更为严峻的不利方向发展。所以，这个主题就应当有着很高的关注度。





但是,目前,社会对于地球的生物科学及环境卫生的未来发展有浓厚的兴趣,因此许多方法被提出来进行预测,可是各有局限。大多数的模型都是基于生物预测方法,它们通过研究一系列健康指标的近期趋势来评估社会经济学的未来发展方向对生物多样性和生态系统服务的影响及作用;其他模型主要应用物种分布的现象学模型,重点研究进化社区生态学的综合规律的潜在作用,并且进一步预测气候变化是如何改变目前观测的地理视距。

上述方法对于预测不同时间尺度下特定健康指标的状态和趋势是比较有用的。但是不足的是,这些方法仍然存在两个重大的弊端:

- (1) 许多研究聚焦于环境与生物系统的当地变化而忽略了复杂的全球变化;
- (2) 它们都忽视了多种环境与生物系统的复杂关系和交叉效应。

因此,建立更为全面且科学的模型成为迫切的需要。

为了完善以上两个不足之处,出题者要求建立一个更好的、基于现有生物预测模型的全球化预测模型,并且,归纳起来能克服以下两个具体的量化建模的挑战:

- ◆ 建立的全球化模型要包含地球上的相互关联系统的复杂性,地区状况对全球系统的影响以及显著的人类作用,并且能够改进生物预测性能。
- ◆ 通过全球化模型同时考虑多个因素能够确定临界点,并且有效地措施能被提出用于阻止和限制临界点的到达。

## 5.5.2 分析与建模及求解

### 一、模型分析

上文进行了问题分析,我们将题目凝练成了2个要求和目标,接下来就要从这两个目标出发,考虑如何去解决这个问题。

首先,从全球网络系统出发,我们需要建立一个全局性的网络模型,要能包含整个地球的生态环境及其关系。但是,这模型里又得包含这每个生态环境内部的多种元素的相互作用和影响,也得考虑不同生态环境之间的元素的相互作用和影响,所以,我们得建立一个多层次的网络模型。其最高层,或者说是最上层就是整个地球的各个生态系统之间的网络模型,接下来一层,又得包含各个元素的作用影响和其性质。

考虑到这,我们决定建立一个双层耦合网络模型,来构建这复杂的生态系统和元素之间的关系,因为上层和下层连接的方式比较复杂、多样化(因为其相互影响的复杂性),所以可以称之为“耦合”。

又因为要求中多次提到这个模型要有能描述生态系统及其元素动态变化的过程,而我们常用的描述动态变化及其影响的模型就是微分方程模型,所以,我们再用相对应的微分方程模型去描述这个双层耦合网络模型中的变化和联系,从而建立起整个复杂的全局模型。

有了这第一部分的问题分析,为了进一步陈述我们的解题方法,接下来,我将介绍这几个部分:

- 在第二部分,我们构建了一个双层耦合网络来分析、预测及评估我们星球的生物





与环境健康状况。并且给出了对全球网络和元素网络的详细描述。

- 在第三部分，我们构造微分方程组来描述网络模型。我们应用平均场动力学 Lotka-Volterra 方法来描述元素之间的相互作用，并且用扩散方程来描述各元素在不同区域之间的扩散传播模式。
- 在第四部分，我们应用建立的模型解决了一系列问题，如地球健康状态的预测和临界点的确定等。同时，详细叙述了灵敏度分析和网络结构的研究。
- 在第五部分，考虑不确定性，我们通过加入小扰动来优化我们的模型。
- 最后，我们详细讨论了模型的优缺点。

## 二、模型建立

### 1. 构建双层耦合网络

对于全球的环境，各种各样的信息可以被有目的地选择，从而形成不同类型的网络。网络中的节点代表反映地球健康状况的元素，可以是国家、洲、海洋、栖息地，或任何其他这些或其他元素的组合；网络中的链接可以代表节点或环境相互之间的影响，或者体现一些物质元素（如污染）随时间的流动或扩散。此外，健康度量也应该被模型所体现。健康度量可以是反映地球状态的任何元素，包括人口统计的、生物的、环境的、社会的、政治的、物理的、或化学的状态，等等。

由于大量的因素和各方面的影响都需要考虑，一个简单的网络不能够很好地反映整个环境与生物系统的状态。一个含多层结构的复杂网络会更加的合理和全面，并且各个网络层之间存在一定的互相作用和依赖关系。因此，我们提出了一个双层耦合的复杂网络，包括全球网络和元素网络，来反应全球状况。

#### (1) 全球网络。

如上所述，我们所挑选的网络节点必须反映地球总体或各地区物理健康状态，同时，也要关注从地区到全球全方位的生物交流和影响。根据地理和气候，我们首先将整个地球划分为几个区域，全球地理与气候图及区域划分如图 5-40 和图 5-41 所示。

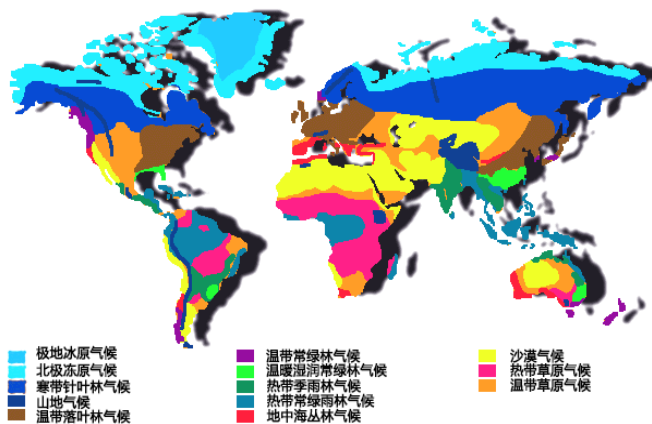


图 5-40



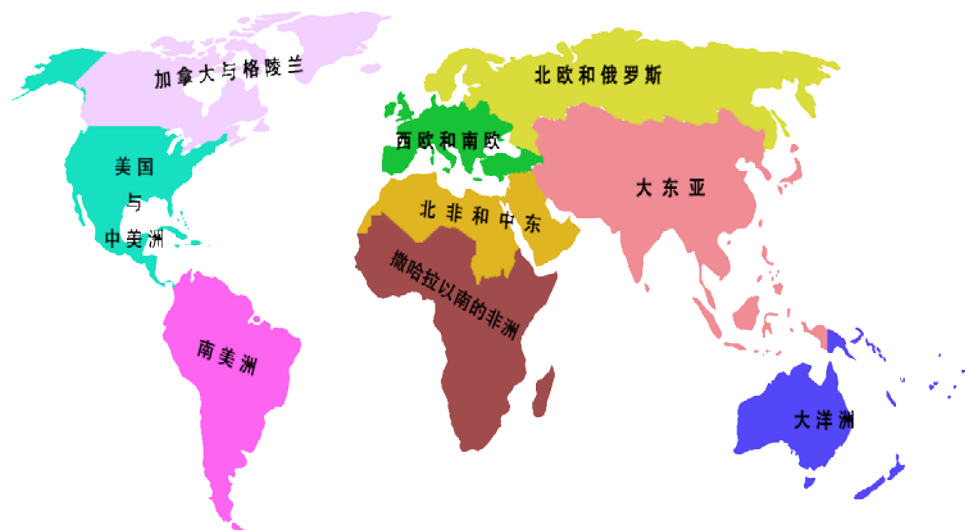


图 5-41

图 5-41 中以不同色彩呈现了地球的平面图。整个地球被划分成九个区域，每一块区域有明显的地理和气候差异，并且用一特定的颜色标记。

我们划分的区域被作为全球网络的节点，并且每两个节点之间的链接，即图 5-42 中显示的实线，代表物质元素随着时间的流动或扩散的路径。实际上，成对的连线一定是大量的，是所有节点的两两组合；但是，考虑到两节点的距离越远，扩散效应越弱这一事实时，我们便只留下每两个相邻节点的直接连接，也就是只考虑任意节点到其相邻节点的连接。图 5-42 展现了全球网络中选择的节点和每个节点之间的直接连接。

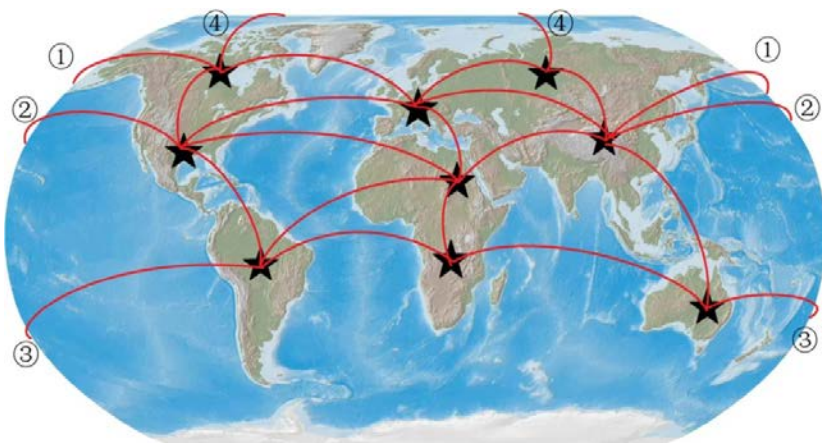


图 5-42

上面显示的地图是平面的，但是，连接是属于球面连接。也就是说，一些连接不能够被很好地平面绘制，因为它们需要绕过东西半球的分界线。因此，我们上图中标出的有同一编号的连接是同一条连接，如图 5-42 中连接①和连接②。为了方便数据查找，每一区域



中的具有代表性的若干个国家（包括领土，领海以及专属经济区）被选择作为我们的研究对象，我们选择的这些国家一般占据了它们所属区域的 70% 以上面积。

### （2）元素网络。

我们建立的双层耦合网络，是由上一节构建的全球网络和这一节将继续构建的元素网络组成。为了评估和预测健康状况，反映地球状况的元素是必不可少的，而这些元素多种多样而且很难确定。

幸运的是，2012 年环境绩效指数（2012 Environmental Performance Index, EPI）基于 10 项政策类别及 22 个性能指标对 132 个国家进行排名。而且，EPI 的方法学追踪各国家在上述类别和指标方面的表现和进展，有利于促进在经济与区域同类团体中的跨国比较。

基于 EPI 数据，我们最终挑选七个具有代表性的元素作为我们研究对象，分别是空气质量、饮用水、生物多样性与栖息地（缩写成生物多样性）、森林、人口密度、气候变化以及能源消耗。毋庸置疑，我们能根据它们的名字得知上述七个指标的含义。而且，耶鲁大学环境法律与政策中心（the Yale Center for Environmental Law and Policy, YCELP）和哥伦比亚大学国际地球科学信息网络中心（the Columbia University Center for International Earth Science Information Network, CIESIN）已经对它们的值规范化。所以，我们选择的元素作为元素网络的节点，也就是说，在这个子网络中有七个节点且连接代表每两个元素之间的联系与相互作用。

因此，根据全球网络和元素网络的分析，我们最终完成双层耦合网络的构建，整体的结构如图 5-43 所示。

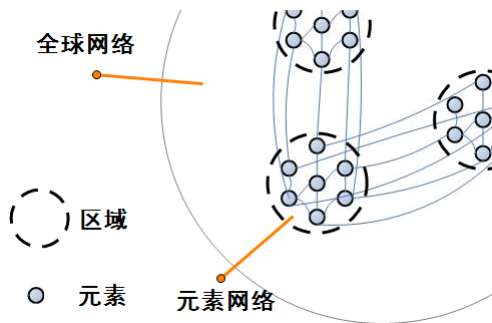


图 5-43

## 2. 基于微分方程组的网络模型

为了预测地球的生物与环境健康状况，我们建立了一个全面的双层耦合网络模型，包括了全球网络和元素网络。此外，上述被选择的元素能够很好地体现地球健康水平。因此，我们认为地球的变化可以通过观测和研究这些元素值的变化来度量。一般而言，元素值的变化有内因和外因之分，这些将会在接下来的文章中阐述。

### （1）内因：元素网络中的变化。

内因也就是元素值变化发生于元素网络中，即任意两元素之间的关联与相互作用。基



于已有的各种理论,我们应用 Lotka-Volterra 方程组来描述这个变化过程。

Lotka-Volterra 方程组,也被称为捕食——食饵模型,是一对一阶非线性微分方程,它通常被用于描述两个种群,一个捕食者,一个被捕食者,相互作用所构成的生物系统的动力学。

我们假设  $u_1(t)$ ,  $u_2(t)$  分别表示两个种群在  $t$  时刻的数量或密度。从考察各自的相对增长率入手,我们通常还需要考虑到种群内自身发展规律和种群间相互作用的影响这两方面。因此, Lotka-Volterra 方程组有如下表达形式:

$$\begin{cases} \frac{du_1}{dt} = u_1(a_1 + b_1u_1 + c_1u_2), \\ \frac{du_2}{dt} = u_2(a_2 + b_2u_1 + c_2u_2), \end{cases}$$

其中,  $a_1, a_2$  分别代表种群  $u_1, u_2$  的无约束自然增长率;  $b_1, b_2$  分别表示种群  $u_1$  影响种群  $u_1, u_2$  自然增长率的影响因子; 同样地,  $c_1, c_2$  表示种群  $u_2$  影响两种群自然增长率的影响因子。

针对于本文问题, Lotka-Volterra 方程组的原理依然可以很好地应用于每个元素的变化。对于给出的七个元素, 全球网络中任意的一个节点就能给出七个方程。此处, 我们以人口密度为例来进行详细描述。

传统上, 一般定义“自然”即为非人类, 但是现在生态环境已经发生了翻天覆地的变化。人类, 作为一个强大的群体, 已经影响了一大片领域。近几十年来人类活动聚集了一系列的产出: 温室气体排放量增加、同温层臭氧的破坏、水层的消耗以及热带雨林的大面积砍伐。这些都削弱了地球自然系统的生产能力(土壤、森林、海洋及生物多样性)或稳定性(气候、海平面及紫外线过滤)。

在当前的生物圈中人类作用已显著增强, 因此, 从某一程度上说, 选择人口密度作为一个元素是不可避免的。显然, 此处我们关注的人口密度对于它本身以及其他元素都有极大的影响。

聚焦于全球网络中的任意一个元素网络, 我们来建立模型。首先, 从人口密度来分析, 我们假设  $X_p^i(t)$  代表  $t$  时刻  $i$  节点处的人口(Population)的密度值。因此, 人口密度的变化可表示为:

$$\left( \frac{d[X_p^i(t)]}{dt} \right)_{IN} = X_p^i(t) \cdot [\gamma_p^i + \alpha_{pp}^i X_p^i(t) + \alpha_{pw}^i X_w^i(t) + \alpha_{pb}^i X_b^i(t) + \alpha_{pf}^i X_f^i(t) + \alpha_{pa}^i X_a^i(t) + \alpha_{pc}^i X_c^i(t) + \alpha_{pe}^i X_e^i(t)],$$

其中,  $A, W, B, F, P, C, E$  分别为七个元素(分别是: 空气质量、饮用水、生物多样性、森林、人口密度、气候变化以及能源消耗)的缩写;  $\gamma_p^i$  为人口密度的无约束自然增长率;  $\alpha_{pp}^i$  表示人口密度对自身的影响因子, 常识告诉我们, 高人口密度会限制自身人口增长, 其实这也是一种负反馈作用;  $\alpha_{pm}^i$  ( $m = A, W, B, F, C, E$ ) 代表其他六种元素对人口密度自



然增长率的影响因子。

同样地，其他六种元素也有类似上述的表达式。由于篇幅所限，我们给出如下形式的通式：

$$\left( \frac{d[X_k^i(t)]}{dt} \right)_{IN} = X_k^i(t) \left[ \gamma_k^i + \sum_m \alpha_{km}^i X_m^i(t) \right],$$

$$k, m = A, W, B, F, P, C, E.$$

(2) 外因：全球网络中的变化。

外因，也就是元素值变化发生于全球网络中，即物质元素的流动或扩散。我们需要先分析清楚其机理，再建立合适的方程和模型来描述这一过程。

#### ① 引用反应——扩散模型的原理

为了研究节点之间物质元素的扩散传播机理，我们通过引用二维反应——扩散模型（R-D equation）的原理来详细描述这一现象。反应——扩散模型常常通过计算区域浓度梯度的散度，并乘上一个扩散系数来表达扩散传播效应。

类似于有从高浓度区域流向低浓度区域能力的流体，扩散传播效应能够用散度很好地来体现。但是，由于节点是离散的，我们必须推导出离散点之间的扩散传播表达式。

一个节点到其相邻节点的距离常常被当作是已知信息，因为距离是很容易测量得到的，而且这距离往往和扩散传播有一定的联系。因此，下一小节提供了一个详细的细节证明来寻找这种联系。

#### ② 基于点源流的证明。

我们假设，存在一个标量场  $f(\rho, \theta)$ ，它的梯度大小为给定点在坐标空间中单位长度的最大  $f$  变化率，即有如下极坐标形式：

$$V = \nabla[f(\rho, \theta)] = \frac{\partial f}{\partial \rho} \bar{e}_\rho + \frac{1}{\rho} \frac{\partial f}{\partial \theta} \bar{e}_\theta$$

因此，标量场  $f(\rho, \theta)$  经过求梯度能转化为一个矢量场  $V$ 。对一源流，所有的流线都是由中心点  $o$  发射出的直线，如图 5-44 所示。明显地，径向方向和切线方向的分量分别为  $\partial f / \partial \rho$  和  $\partial f / \partial \theta$ ，此处， $\partial f / \partial \theta = 0$ 。

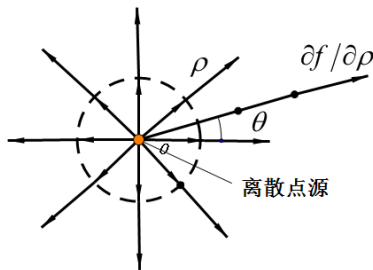


图 5-44

根据反应——扩散模型的机理，我们很容易得到矢量场  $V$  中每个点的散度：

$$\begin{aligned}\operatorname{div} V &= \operatorname{div}(\nabla f) = \operatorname{div}\left(\frac{\partial f}{\partial \rho} \bar{e}_\rho\right) \\ &= \frac{1}{\rho} \frac{\partial}{\partial \rho} \left( \rho \cdot \frac{\partial f}{\partial \rho} \right) = \frac{1}{\rho} \frac{\partial f}{\partial \rho} + \frac{\partial^2 f}{\partial \rho^2}\end{aligned}$$

其中， $\operatorname{div}(\bullet)$  代表散度而  $\nabla(\bullet)$  代表梯度。

上述公式右端中的第二项，也就是  $\partial^2 f / \partial \rho^2$ ，对最终结果的影响微小。因此，为了便于计算，我们忽略此项。则结果为：

$$\operatorname{div}(\nabla f) \propto \frac{1}{\rho}$$

最后的结论为，标量场的梯度的散度与径向距离成反比。

### ③ 扩散传播方程组。

根据上一节得到的结论，同时综合反应——扩散模型（R-D equation）的机理，可知离散点之间元素的扩散传播是与它们之间的距离成反比的。

假设元素  $k$  从节点  $i$  扩散到节点  $j$ ，由于我们的假设，节点  $j$  必定与节点  $i$  相邻。并且元素  $k$  的变化量可以表示为：

$$\delta = \beta_k^i \cdot \frac{X_k^i(t) - X_k^j(t)}{R_{ij}}, \quad k = A, W, B, F, P, C, E.$$

其中， $R_{ij}$  为节点  $j$  到节点  $i$  的距离，其所有的值可在谷歌地图上测量获得；而  $\beta_k^i$  为节点  $i$  的传播因子，为我们所设的参数。

显然，节点  $i$  的所有相邻节点都对元素  $k$  的扩散传播有贡献。元素的变化量取决于相邻节点数。对于七个给定元素和九个节点，存在一系列的方程，为了书写简洁，我们给出如下的通式：

$$\left( \frac{d[X_k^i(t)]}{dt} \right)_{EX} = \sum_j \left[ \beta_k^i \cdot \frac{X_k^i(t) - X_k^j(t)}{R_{ij}} \right]$$

$$k = A, W, B, F, P, C, E.$$

其中，节点  $j$  必须为节点  $i$  的相邻节点。

但是，对于人口密度这一元素，它的值并没有同其他六个元素一样根据 YCELP 和 CIESIN 提出的规则进行规范化处理。而从分析来看，人口密度越大，它的值越小。换句话说，人口密度这一元素的扩散传播与其他六个元素不同。





### 三、模型求解

#### 1. 联立求解与参数拟合

实际上，双层耦合网络中的全球网络和元素网络常常相互作用相互影响，甚至存在紧密的联系与反馈。因此，我们需要进行联立求解。

根据前几节的分析与结果，我们发现元素值的变化等同于由内因和外因共同作用的总变化。所以，我们最终的双层耦合网络模型能被描述为以下形式：

$$\begin{aligned}\frac{d[X_k^i(t)]}{dt} &= \left( \frac{d[X_k^i(t)]}{dt} \right)_{IN} + \left( \frac{d[X_k^i(t)]}{dt} \right)_{EX} \\ &= X_k^i(t) \left[ \gamma_k^i + \sum_m \alpha_{km}^i X_m^i(t) \right] + \sum_j \left[ \beta_k^j \cdot \frac{X_k^i(t) - X_k^j(t)}{R_{ij}} \right] \\ k, m &= A, W, B, F, P, C, E.\end{aligned}$$

参考所给元素的十年数据，我们采用最小二乘法（OLS）对之前建立的参数方程组进行参数拟合。应用 MatLab 编程计算后，我们便得到了其最后结果，并在下一部分进行应用和分析。

#### 2. 应用与分析

经过参数拟合之后，所有参数值就被确定，我们模型也最终完整。接下来，通过应用我们模型，我们重点解决相关问题以及给出分析。

##### （1）预测地球健康变化及确定临界点。

全球健康变化的预测是我们的主要任务之一，而临界点是一个非常重要且必不可少的指标用于呈现这种变化。

##### ① 健康变化的预测。

根据我们所构建的模型和查找到的数据，我们首先预测每一区域每一元素在未来五十年的数值与变化趋势。此处有九块区域，因此，由于篇幅所限，我们选择两个具有代表性的区域，分别是美国和撒哈拉以南非洲所属区域，作为我们的研究对象进行分析和论述。此处，我们标记这两个区域为 U 区域和 S 区域。

图 5-45 和图 5-46 所示分别反映了在未来五十年 U 区域与 S 区域中每一元素的变化趋势。很明显，在 U 区域中，森林，饮用水和生物多样性这三个元素的评分急剧增长，然而剩下的四个元素几乎没有变化。也就是说，在合理的政策下，U 区域的生态系统是健康的。另一方面，S 区域是不发达区域，但是由于人口的增长，丰富的自然资源被需求用来改善生活质量。因此，过度开采导致了气候恶化，尽管其他元素值仍保持增长的。

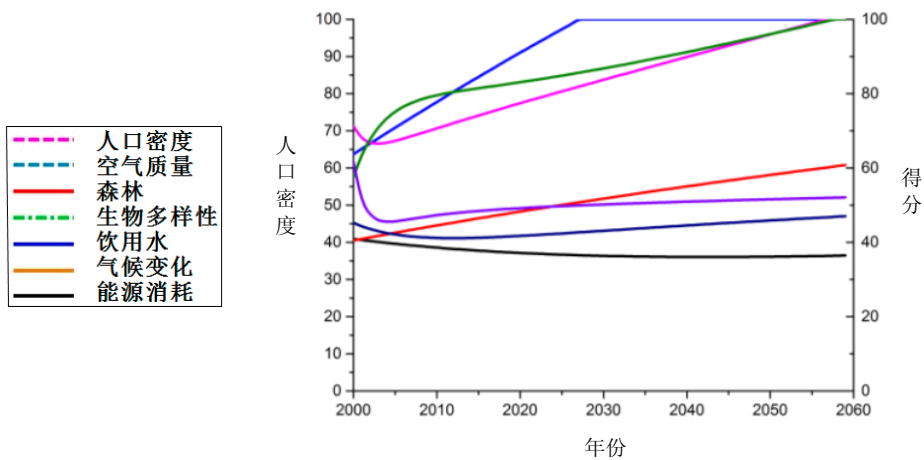


图 5-45

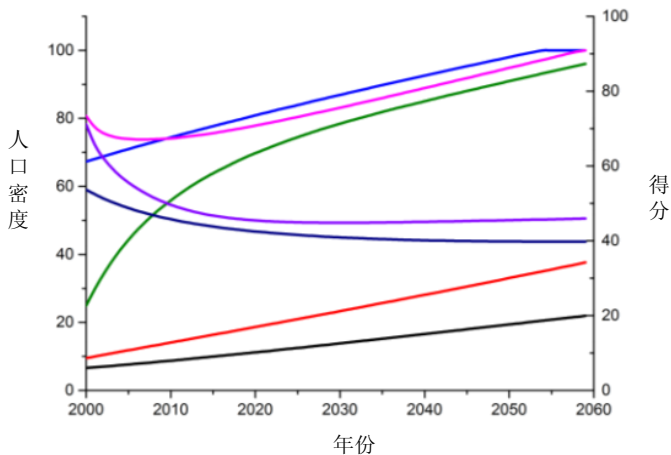


图 5-46

② 确定临界点。

临界点是一个引爆点或转折点，且驱使地球生物圈发生不可逆转的变化。换句话说，在达到临界点之前，我们人类还有机会采取有效的措施来控制局面；而一旦过了临界点，则将无法挽回。

以空气质量这一元素为例来详细说明。以寻找空气质量的临界点为目的，我们首先保持其他六个元素的初始值不变，从高到低不断改变空气质量元素的初始值。接着，我们的模型将会计算出未来五十年内空气质量的变化趋势，进而来判断这个变化趋势是否是不可逆的一直变坏。当空气质量值第一次出现长时间段的降低趋势且没有任何回复迹象时，这个值即为我们所寻找的临界值。





利用 U 区域内所有元素的数据, 我们可以根据上述思路来寻找这些元素的临界点。而且我们发现七个元素中只有两个元素, 空气质量和生物多样性, 有临界点。当到达临界点时, 这两个元素的变化趋势与其对应的原始趋势的比较如图 5-47 和图 5-48 所示。

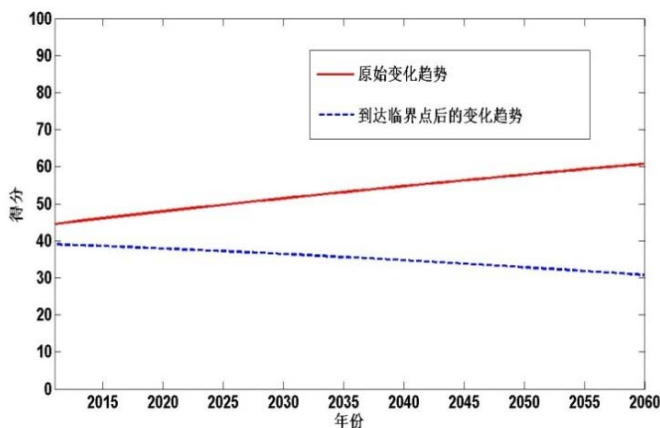


图 5-47

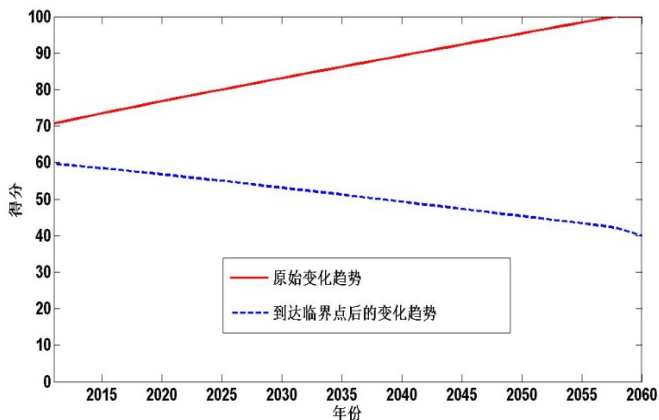


图 5-48

由图 5-47 和图 5-48 可以看出, 一旦到达对应的临界点, 空气质量与生物多样性的元素值是保持下降趋势的。且得到空气质量和生物多样性的临界值分别为 38.76 和 59.86。常理告诉我们, 这种现象是不利于全球健康的。因此, 我们建议 U 区域的政府和决策者需要密切关注空气质量和生物多样性, 并且需要采取有效的措施来预防临界点发生。

## (2) 灵敏度分析与人类作用的重要性。

众所周知, 人类, 作为一个强势群体, 已经对大片的领域造成影响。甚至, 在目前生物圈中人类作用与影响显著增强。在本节中, 我们将重点放在研究人类作用的重要性上, 研究人口密度对其他六个元素的影响, 并对我们的模型进行灵敏度分析。即我们将会研究人口密度对其他六个元素的影响。由于篇幅所限, 我们选择两个具有代表性的影响因子做详细叙述。

图 5-49 和图 5-50 分别展现了人口密度对森林和能量消耗的自然增长率作用的影响因子的灵敏度分析的示意图。在图中，实线代表元素的原始变化趋势，虚线代表元素调整后的变化趋势。

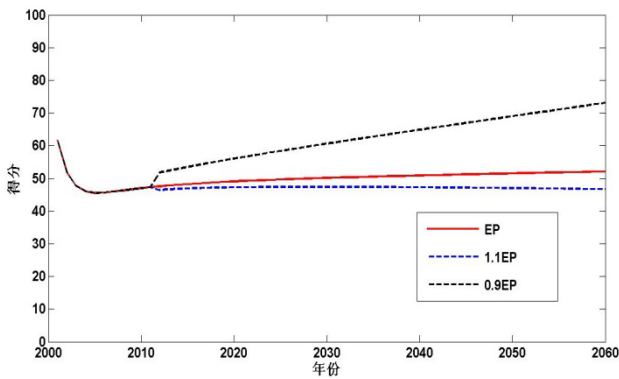


图 5-49

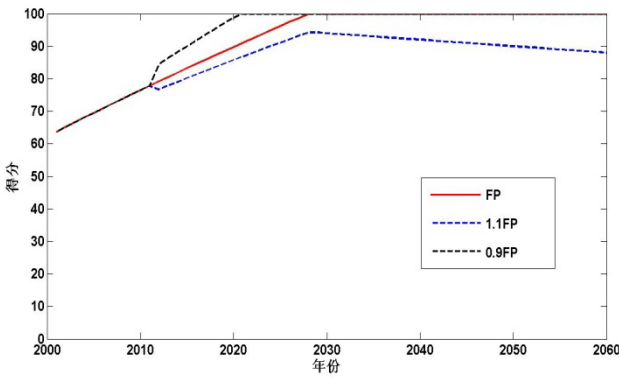


图 5-50

注：FP 和 EP 分别是人口密度对森林和能量消耗的自然增长率作用的影响因子。

显然地，EP 这一影响因子的原始作用几乎保持不变。也就是，EP 是一个稳定的参数，且能被较好地控制。此外，减弱 EP 因子的影响作用只会产生微小的变动，而增强 EP 因子的影响作用能改善地球健康状态。与 EP 因子相对比，FP 影响因子在未来发展中扮演了一个较为重要的角色。增强 FP 因子的影响作用能促进健康状况的改善，然而减弱 FP 因子的影响作用看起来更为复杂。在减弱 FP 影响作用的初期，FP 的作用变化是微弱的，但是在下一阶段，FP 就产生了较为明显的负作用。

(3) 分析网络结构。

根据平均场动力学 Lotka-Volterra 模型中经参数拟合得到的参数数据，我们可以通过比较各个参数值大小进而来判断这些参数互相影响的强度及正负作用。设定这种相互影响强度的平均绝对值的十分之一为限定值，我们认为数值小于此限定值的参数对整体的影响很小，可以忽略。又由于各个地区的参数值不同，我们可以给出各个区域的所有元素的影响





网络。

以 U 区域（即美国所在区域）为例，此区域的元素影响网络的结构图如图 5-51 所示。

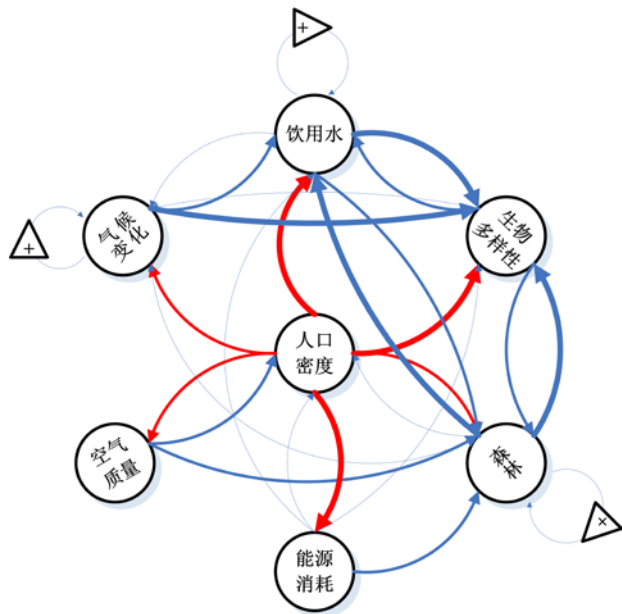


图 5-51

在图 5-51 中，线条的粗细代表了影响因子值大小，即影响其他元素的能力。然而三角形表示自身的正负反馈。

根据上述的影响网络图，我们可以进行网络拓扑结构分析和带权值的网络分析。如表 5-14 和表 5-15 所示。总体而言，存在一些广泛用于表现网络特征的度量。此处，我们使用四个常用的基本度量：

“点出度 (Out-Degree)”：点出度指的是某节点直接连向其他节点的链接数；

“点入度 (In-Degree)”：点入度指的是直接连向某节点的链接数；

“局部中心度 (Local Centrality)”：局部中心度数值上等于直接连向某节点的链接总数；

“整体中心度 (Global Centrality)”：整体中心度数值上等于其他节点连向某节点的捷径（两点之间最短距离称之为捷径，geodesic）总数。

一般而言，局部中心度和整体中心度都可以找出处在一系列联系的“核心”位置的点，此点称之为中心点 (Central Point)。由于我们的网络是一个只有七个节点的小网络，而且任何两个节点之间的距离不会超过 2，因此，用局部中心度就已经可以找出网络的“核心”节点了。

另外，考虑到这是一个带权重的联系网络，衡量一个节点（即一个元素）的重要性的较佳办法就是衡量这个节点对这整个网络的影响贡献，而每个节点的出度 (Out-Degree) 表示了它对其他节点的影响，而其权值则表示它对其他节点的影响强度。所以，我们便可以定义一个节点对它所在的网络的影响能力，我们称其为“影响度” (Influence Degree)。





其数值上等于这个节点所有出度上的权值的平均绝对值（考虑到权值存在正负的影响，所以取绝对值）。

表 5-14

元素	中心度
人口密度	9
空气质量	3
森林	10
生物多样性	6
饮用水	8
气候变化	7
能源消耗	5

表 5-15

元素	影响度
人口密度	7.93
空气质量	3.35
森林	4.65
生物多样性	2.56
饮用水	6.08
气候变化	5.14
能源消耗	1.48

表 5-14 和表 5-15 显示了基于网络拓扑结构分析和带权值的网络分析的结果。在网络的拓扑结构的分析中（见表 5-14），我们发现森林这一元素为这个小网络里联系最多的中心点，而人口密度屈居第二，饮用水居第三位；但在带权值的网络分析中（见表 5-15），我们发现人口密度对网络的影响最为巨大，远远高于屈居第二的饮用水资源，而森林位于第四位。对比这两个结果，很显然排名是不一样的。换句话说，这也在一定程度上体现了生态网络环境的中心元素与地球上影响作用最大的元素之间的差别。

3. 模型优化——加入小扰动

常理告诉我们，现实生物与环境系统中存在大量的不确定性与随机性，因此系统的发展似乎很难预测。此处，为了更好地描述我们的全球耦合网络，我们在模型中针对每一参数加入服从标准高斯分布的小扰动来显示随机性。因此，加入小扰动后的双层耦合网络模型可有如下形式的表达式：

$$\begin{aligned} \frac{d[X_k^i(t)]}{dt} = & X_k^i(t) \left[ \left( \gamma_k^i + \hat{\gamma}_k^i \right) + \sum_m \left( \alpha_{km}^i + \hat{\alpha}_{km}^i \right) X_m^i(t) \right] \\ & + \sum_j \left[ \left( \beta_k^i + \hat{\beta}_k^i \right) \cdot \frac{X_k^i(t) - X_k^j(t)}{R_{ij}} \right] \\ & k, m = A, W, B, F, P, C, E. \end{aligned}$$

其中，上标  $\wedge$  表示参数的小扰动增量；节点  $j$  必须为节点  $i$  的相邻节点。

图 5-52 所示为在加入小扰动后所有元素变化趋势，显示了在加入小扰动之后所有元素的变化趋势。与原始结果进行对比，加入小扰动后得到的结果会发生一定的或大或小的变化，也会更难预测元素值的变化趋势。

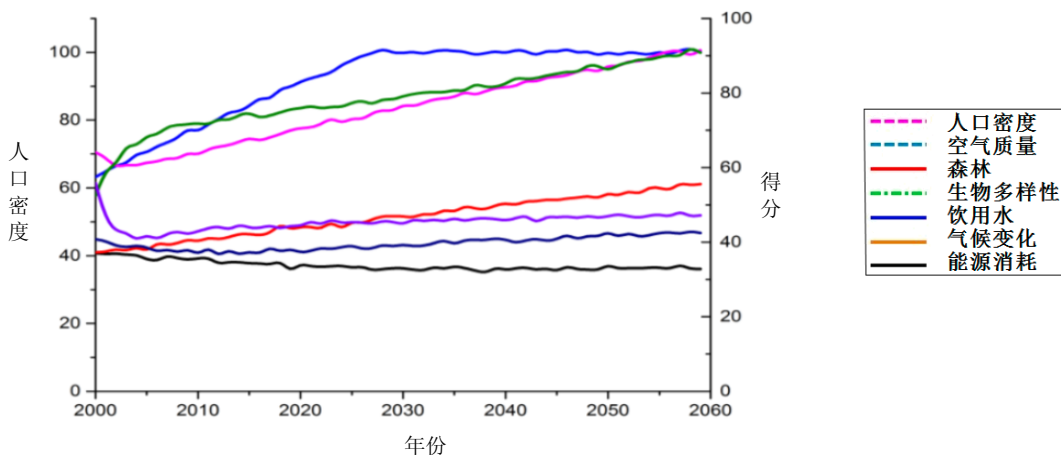


图 5-52

### 5.5.3 论文点评

#### 一、优点

一篇合格的文章，在至少满足题给要求外，还得有足够的新颖可以吸引人的眼球，对于这题，本文有以下几点值得回味：

##### 1. 全球性

考虑到不同方面的七个元素并将全球划分为九个区域，我们的模型从某种程度上而言已经较为全面。此外，如果有各个区域更为详细的数据，我们就可以丰富模型考虑的元素数量，使模型更加完善，以至于能尽可能体现复杂的全球网络。甚至，如果所有国家的数据被给出，我们还可以把区域细化到各个国家，从而更加精细地展现全球的变化。

##### 2. 复杂性

我们的模型展现了地球相关生态系统的复杂性，包括多元互动、正负反馈环、突发性行为以及即将发生的状态变化或临界点，并且充分描述了反映健康状态的元素之间的联系与相互作用以及地球生态系统的行为。

##### 3. 科学的评分

我们的模型与解法针对区域和国家的健康水平给出了客观科学的评分，也为政府和决策者提供了可供参考的信息和建议。此外，根据这个评分，我们可以拟定一个排名，显示了各地区和整个地球的生态系统的健康状况，从而有了更加直观的了解，并以此作为政策

制定和修改的信息依据。

#### 4. 可视化

由于我们模型描述的对象比较全面，所以其所得结果也非常丰富，而且模型是个随时间变化的动态模型，当生成随时间变化各元素的变化趋势图时，我们便可以直观地观察各个元素的长远影响和变化。此外，在研究元素间的相互作用影响以及各元素在不同区域间的传播扩散时，我们通过分析双层耦合网络的构造来展现其中的重要元素及其影响能力。

#### 二、缺点

- ◆ 由于我们的模型是一个复杂的双层耦合网络模型，复杂的网络和更丰富的数据必然会导致一个不可避免的现象——计算量成倍增加，从而影响模型求解的效率。
- ◆ 在分析元素间的联系与相互作用时，我们采用数据导出的线性关系来替代实际机理，因此，我们模型得出的结论可能会与实际有一定的差异，即可能失去了一定的真实性。

#### 三、总结

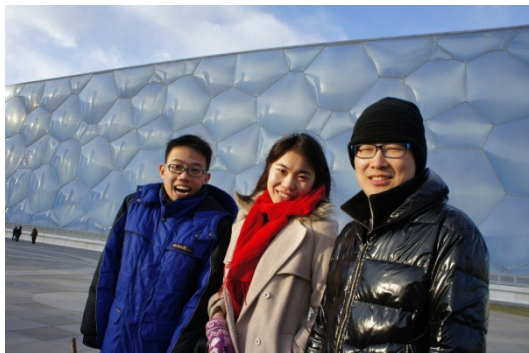
其实能够获得 Outstanding 奖，也实属侥幸。我们三人于 2012 年第一次参加国际数模，最后的成绩是 Meritorious Winner。在 2013 年，我们都已经大四快结束了的时候，又一起参加了最后能参加的美赛。四天结束后，我们的第一感觉是，至少从时间匆忙程度、论文内容及排版和模型覆盖面上来说，都要比上一年的要好。待到成绩揭榜，听说我们是 Outstanding Winner，也真的高兴了好几天，我们三人的数模生涯终于完整了。

说起这段回忆，一方面是想让读者能够了解这样三个名字，舒毅潇、王宇辉、陈超，能够想象到我们三人从大二开始一直坚持到大四的困难和勇气。

另一方面，是想用我的经历告诉各位读者：

数模是一项值得你投入时间精力的事业，如果你真心喜欢它，就请全力投入其中吧，只要你一直用心学习，就一定能不断提高，最后，也一定能收获你们的全满！与君共勉！

图 5-53～图 5-57 所示是我们这个团队与队员照片及 2013 年参赛获奖证书。



团队合影，时隔 2 年于北京重聚时留念（从左到右为王宇辉、陈超、舒毅潇）

图 5-53



建模队员，本文作者，舒毅潇  
现为西北工业大学航空宇航制造工程专业研究生

图 5-54



编程队员，王宇辉  
现为中国科学院计算所研究生

图 5-55



写作队员，陈超  
现为西北工业大学飞行器设计专业研究生

图 5-56



图 5-57







## 第6章 经验分享：做一名成功的指导者和参赛者

数学建模竞赛不单单是能力的大比拼，也是参赛经验的较量。本章通过具备多年数模指导经验的老师以及多次参赛经验的学生的自述，向读者传授数学建模的指导经验和参赛经验。

### 6.1 优秀指导教师讲数模

“全国大学生数学建模竞赛优秀指导教师”大连大学谭欣欣老师、西北工业大学“模教教主”肖华勇老师都具备 15 年以上的数学建模指导经历，曾经多次带领队伍参加全国大学生数学建模竞赛以及美国大学生数学建模竞赛，获奖无数。他们用自己的亲身经历讲述如何成为一名成功的数学建模指导教师。

#### 6.1.1 数学建模活动是培养大学生创新能力的有效途径

我是大连大学的数模指导教师谭欣欣，我的数学建模历程起始于 1998 年，至今已有 16 年了。1999 年，第一次组织学生参加全国大学生数学建模竞赛，5 个代表队，只有一名指导教师，但却成为大连大学数学建模活动的卷首和开篇。首战告捷，给了全校师生巨大的鼓舞。

至今我们已连续 16 年组织学生参加全国大学生数学建模竞赛，连续 13 年组织学生参加国际大学生数学建模竞赛。为使更多的学生了解数学建模，参与数学建模，进而受益于数学建模，2000 年起，我校又开展了“大连大学数学建模竞赛”。总计先后有 3 万余人参加了各类数学建模课程学习、国内外数学建模培训及竞赛以及数学建模工作室活动。累计获得 111 个国际奖；42 个队获得全国奖；180 个队获辽宁省奖。无论是受益面还是竞赛成绩均位于辽宁省普通高校前列。“一次参赛，终生受益”已成为所有参赛同学的共识。

自 1998 年大连大学第一批教改项目——“数学建模的教学与实践”的启动，我们开始迈入了数学建模的教学与实践的领域。由于数学建模教学与系列实践活动涉及的知识领域广泛，在地方综合性大学中对学生进行知识、能力、素质培养中发挥的独特作用，得到广大学生的认同与喜爱、学校各级领导的重视与支持。目前，我校数学建模教学与实践已步入系列课程建设——学生踊跃参与——人才培养效果显著——贯穿大学教育的全过程，从而形成数学建模教学与实践更上一层楼的良性循环。





## 1. 建立数学建模教学体系，探索教育教学模式改革

在多年的实践探索中，结合学校实际，逐步建立起数学建模教学体系，整体设计课内外、校内外教学活动，以此推动传统的教育教学模式的改革。

数学建模教学体系与框架如图 6-1 所示。

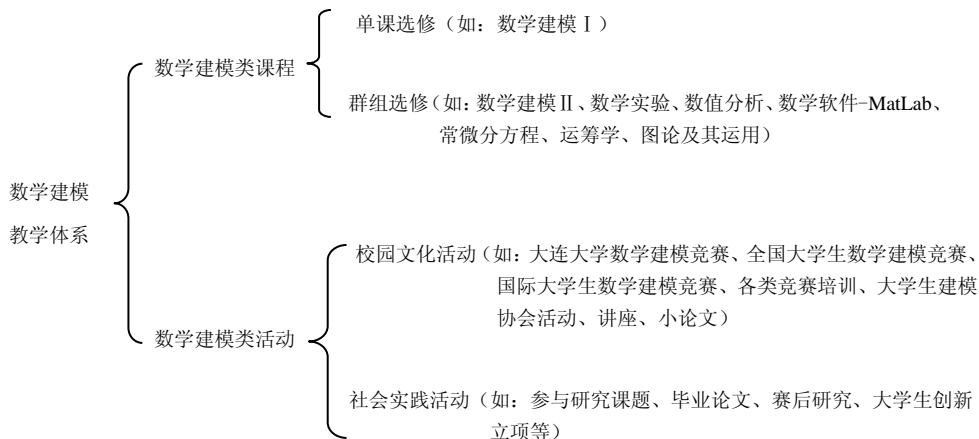


图 6-1

### ◆ 课程体系层次丰富，课程内容结合实际

针对不同专业的学生，开设既有共性又各具特色的不同类别的数学建模课程，包括：本科数学类专业、本科理工类专业、本科管理类专业等，满足不同学生的需求。结合竞赛，每年开设数学建模普及性讲座 3~4 次，每次 2~4 周，主要面向 1 年级学生；对 2~3 年级学生，开设多个层次的数学建模必修课和全校性公共选修课，并将数学建模的思想渗透到如“常微分方程”、“概率论与数理统计”、“高等数学”等课程中；还在数学教育、信息与计算科学、计算机教育等专业，精选数学建模类问题，在学生的毕业论文中继续研究。

根据教学与实践积累，编写了“导航——数学建模教程”作为数学建模类课程的主要教材，除了介绍数学建模的基本方法外，还结合社会现实，选择热点问题作为每一章的案例精选：结合北约与南联盟之战的实际，讲授了战争模型、核武器竞赛问题；结合我国长江防汛，讲授了大坝的空洞探测问题；以及如何预报人口的增长、公平的席位分配、森林救火等。并不断更新教学内容，近年又增加了 SARS 流行病、HINI、AIDS 等预测、航空公司的超员订票、居民小区的治安防范以及大学生学习水平的评判、公务员招聘的双向选择、酒后驾车等，使学生将抽象的数学理论与生动的实际问题联系起来，促进他们更好地应用数学、品位数学、理解数学和热爱数学，激发学生参与探索的兴趣和学习数学解决实际问题的愿望，在知识、能力及素质三方面迅速的成长。

### ◆ 教学方法生动活泼，课程考核形式多样

我们还在每一单元加一次专题讨论课，讨论课由学生主讲。题目在平时讲课时布置，适当引入校园实际问题，如：拥挤的水房、合法容量、大学生学习水平测评等，建立了“以



教师为主导，学生为主体”的实践教学方式，形成师生互动，学生互动的氛围。我们还将每年一次的校内数学建模竞赛的讲座、撰写论文作为学生数学建模课程课堂实践内容之一，并将所提交的竞赛论文和平日的课堂讨论一起作为总成绩的一部分。这样，课堂讲授、问题讨论与课外练习并重，给学生一个具体的实战机会去参与解决实际问题的全过程。特别讨论式的教学、使学生变被动学习为主动学习，调动了学生的主观能动性。这样的学习方式，得到了越来越多同学的欢迎。

### 2. 建立数学建模实践基地，为大学生的创新能力培养提供平台，培养大学生综合素质

我校将数学建模教学与实践活动列入学校创新教育工程中，贯穿于大学教育的全过程。学校拨专款建立数学建模实践基地——数学建模工作室，为开展大学生课外科技活动、促进学生的个性发展、培养创新型人才提供了条件和保障。现在，它已成为国家级的大学生创新实践活动基地。在我校的创新型、应用型人才培养中起到了不可低估的作用。

经过长期的学习我发现，指导教师同学生一起进行赛后研究，对学生的能力提高是最快的，就常年坚持这一作法。在2014年的一年内，工作室就组织针对全校的国内外数学建模赛题研讨40多场。目前，包括“A Cost-Sharing Model of Agricultural Disaster Insurance”、“Prediction Model for Law Enforcement Agencies on the Serial Criminals”、“网络排名算法在提高个人影响力方面的应用”、“基于模拟退火算法的碎纸片拼接技术研究”在内的18篇论文已在《大连理工大学学报》等全国核心期刊上发表。同时，我们还将元胞自动机、排队论等专题研讨、讲座内容整理分类，所有活动及资料面向全校同学开放。

工作室还长年举办数学建模讨论班活动。以三大竞赛为契机，数学建模工作室面向全校，常年举办数学建模讲座。每年四月，东北联赛之前，工作室同学分别到信息、机械、经管、建工、医学等8大学院针对全体新生进行数学建模宣传讲座活动。之后，再面向全校举办数学建模讲座。我们精心设计讲座内容，让一年级新生在学完微积分的基础上能听得懂。目前正在进行的讲座内容就包括：统计回归、最优化理论、评价方法以及微分方程在数学建模中的应用，科技论文文献检索、排版，如何撰写数学建模竞赛论文，数学软件MatLab、Lingo初步及应用，蒙特卡洛模拟算法以及《高等教育学费标准探讨》等5篇数学建模优秀论文赏析。每年，数学建模工作室都面向全校举办这样的讲座30余场。晚上7:00—9:00，你常常可以看到容纳400余人的电教001教室场场爆满。

我校拥有11大学科门类，61个本科专业，数学建模工作室吸纳了来自理、工、医、管理、经济、教育各学科的同学参与。而且数学建模涉及自然界与社会经济各方面千姿百态的实际问题，来自不同专业的同学在一起交流、碰撞，延伸了大学生的知识结构。在数学建模教学实践过程中，我们敢于打破常规，不受习俗束缚，勇于独辟蹊径，使工作室不仅作为知识传授的场所，而且成为各种新观念、新想法萌生、涌动、交流和碰撞的园地。学生们说：这种思维碰撞所带来的快乐是无穷的。剑桥大学许多诺贝尔奖不正是教授们在一起喝下午茶时，思维碰撞出来的吗？三个人依据自己的思维提出设想，有时你不得不为队友的创造性思路兴奋不已。数学建模工作室已成为我校大学生课外科技活动的重要场所。以数学建模工作室为基地，教师带领学生这种小组讨论式的研究，张扬了学生的个性、激





发了学生的潜能, 让学生的精神世界的独特性得到尊重, 使他们不仅在实际问题研究中得到数学科学素质的训练和各种能力的提高, 而且更重要的是得到数学文化精神的陶冶和启迪, 在科学真理与完善人格两方面得到收获。

近年来, 数学建模工作室的学生已在国内核心期刊发表论文 20 余篇。并有多人获辽宁省三好学生标兵、辽宁省高等学校优秀毕业生、“挑战杯”辽宁省创业设计大赛金奖、“东软杯”中国大学生软件之星大赛奖等称号。

2003 年, 我校 6 名学生获教育部首批“国家一等奖学金”, 其中 3 名学生来自数学建模工作室。2004 年, 大连大学首届荣获“毕业生十佳创新奖”的 10 人中, 有 7 名学生来自数学建模工作室; 2008 年数学建模工作室 9 名毕业生中, 6 名学生分别考取了中国药品生物制品检定所、大连理工大学、成都电子科技大学等硕士研究生; 从 2004 年至 2014 年的十年, 大连大学共有 100 名毕业生荣获“大连大学十佳创新奖”, 其中 42 名学生来自数学建模工作室。

事实表明: 以工作室为基地, 以数学建模活动为载体的创新教育模式, 提高了学生的综合素质。不仅培养了大批应用型人才, 使众多学生受益, 同时使大批优秀学生脱颖而出。通过几年的努力, 数学建模工作室这一素质教育实践基地已成为培养高素质、创新型大学生的摇篮。

### 3. 我的体会

#### ◆ 数学建模活动使学生、学校、教师三方共赢

第一, 这项活动培养了学生综合运用知识的能力和创新精神。将通常一门课一门课地孤立的学习, 转化为用数学、计算机以及各个实际应用领域中的知识, 融合起来来完成题目的资料收集、方案论证和撰写论文。这样的培养模式在正常的大学生学习阶段是不多见的。数学建模问题大都是以前没有碰到过的实际问题, 同学们参加数模竞赛时, 他们必须开动脑筋、拓宽思路, 充分发挥创造力和想象力, 因此, 数学建模培养了学生的创新精神; 同时, 这项活动培养了学生的团结合作精神与相互协调的能力。在学校里学生通常是自己一个人看书和做题, 几个人在一起活动的机会并不多, 特别是不同专业的学生在一起研究讨论问题的机会就更少了, 而数学建模要三个同学共同完成一篇论文! 他们在研究与竞赛中要互相讨论、分工合作, 必然既有相互启发、相互学习, 也有相互争论, 这就需要同学们学会相互协调, 求同存异, 这种团队精神与协调能力在他们毕业后的工作中, 以至对于他们一生的发展都是非常重要的。

第二, 通过数学建模所进行的大学生创新精神、综合素质的培养是一种学生和教师间的双向活动。通过数学建模教学与竞赛, 我们把学生生命中的探索欲望燃烧起来, 创造的潜能释放出来。我们在付出的同时, 接受对自己能力与人格的挑战, 体味着与学生共同成长的欢悦, 共同收获的满足。在促进教师水平的提高和知识面扩大的同时, 造就了一批有奉献精神、勇于探索教学改革新思路的师资队伍。指导教师从最初只有我 1 人, 发展到今天已有 5 名, 其中 3 位已完成了在读博士的学习; 共主持、参与国家自然科学基金、省科研项目、教改项目等 19 项; 发表论文 50 余篇; 编写了由高等教育出版社与施普林格出版

## 第6章 经验分享：做一名成功的指导者和参赛者

社联合出版的《实验微积分》等7部著作；获“全国大学生数学建模竞赛优秀指导教师”、大连市人民政府颁发的“科学技术进步二等奖”、“课堂教学最受学生欢迎的教师”等荣誉称号28项；编写了“导航——数学建模竞赛教程”、“导航——数学建模教程”等5部讲义。

从事数学建模教学和竞赛工作十几年来，有拼搏的快乐，也有奋斗的艰辛。为使更多的学生受益于数学建模，使其素质及能力得到提高，无论数学建模工作室的研讨还是国内外竞赛的讲座、辅导，我都安排在晚上6点以后。16年来，年年如此；每年有3次竞赛，次次如此。指导教师在阶梯教室、数学建模工作室和学生们切磋、交流，常常到晚上九点多钟，学生们还不肯离去。这样方便了学生，可老师们回到家里，已是晚上十点钟了。看到老师为了他们如此辛苦，学生们感到了老师对他们的爱，懂得了回报。为我校夺得国际竞赛一等奖的物理系04届董翔宇同学，毕业近十年来，每次竞赛前都主动回到学校，在数学建模工作室带领低年级同学进行课题研究，辅导MatLab软件；在华锐风电任现场经理兼项目经理的李高强，把一年的年假集中到国际竞赛前的寒假，回校培训学弟学妹；工商管理专业毕业的徐鹏同学，推迟了去软件公司就业的时间，毕业论文答辩之后送给工作室一个礼物——校内数学建模竞赛网上报名系统和数学建模工作室网站。徐鹏同学懂得：是大连大学，是大连大学的数学建模工作室，使他由一个文科学生，掌握了连理科学生都望尘莫及的多项技能，并在《东北财经大学学报》上发表了用元胞自动机进行“虚拟经济对我国经济影响的量化研究”论文；计算机科学与技术专业刘奇奇同学，利用业余时间撰写了“元胞自动机学习及编程实例简要教程”、神经网络算法研究，作为工作室的研究资料。正在中国科学院攻读硕士学位的周吕文同学，毕业后一直都在通过网络辅导工作室的学弟学妹们……

最令我们欣慰的是数学建模课程、数学建模竞赛对学生能力的培养，对他们一生有如此大的影响，这是我们始料不及的。借此机会，我要感谢多年来一起拼搏奋斗的同学，是他们对知识的渴望、对数学建模活动的热情，督促着我一直坚守在数学建模工作第一线。

第三，我们立足于大连大学，辐射于省内其他高校。我们编写的著作、教程及光盘等使许多兄弟院校受益；曾被邀请到多所省内高校进行专题讲座；中国教育报、全国大学生数学建模竞赛通信、大连晚报、大连开发区报、大连大学报三十余次报道了我校有关数学建模教学与竞赛的信息；在第八届全国数学建模教学及应用会议上交流了我校开展数学建模教学与实践的经验；我也多次代表辽宁赛区出席全国数学建模工作及颁奖大会。原大连大学校长赵亚平为数学建模活动题词：综合各学科知识，感受实际，体验创造，健全个性——数学建模活动对大连大学的综合素质教育功不可没！

全国数学建模组委会秘书长、清华大学姜启源教授评价我校的数学建模教改项目：“基础扎实，成果突出，效果显著，在数学建模教学与实践改革上特色鲜明，该项目对高等院校特别是地方高等院校探索大学生创新教育的有效途径，具有明显的示范作用”。大连大学的数学建模扩大了大连大学在国内外的影响。

- ◆ 学校的支持是数学建模活动得以顺利进行的保障

大连大学能够在国内外大学生课外活动最重大的赛事中榜上有名，培养了大批优秀学





生, 我校数学建模的受益面也跃居东北地区地方高校前列, 这决不仅仅是几个人努力的结果。大连大学良好的文化氛围, 优良的制度体系, 是数学建模教学与实践得以顺利进行的保障。校领导的远见卓识, 积极支持, 同学们的踊跃参与, 激发了指导教师忘我的热情以及为人民教育事业努力工作的责任感。忘不了, 每年大年三十, 电话里传来校长的亲切问候, 以及教务处领导的探望和关怀, 在寒冷的严冬, 给参加国际竞赛的师生送来暖暖春意, 使我们备受鼓舞。忘不了, 信息工程学院为参赛师生准备了最好的计算机及场地, 为指导教师的学习及工作提供了极大的便利。借此机会, 我们深表感激。能够用辛勤的劳动为教育改革、培养 21 世纪高素质人才以及大连大学的腾飞发挥一点点光和热, 我们感到莫大的欣慰和荣光。

我只是做了一个教师应该做的事, 但却得到了许多荣誉: 大连大学教学名师、大连市劳动模范、全国大学生优秀指导教师……我深知, 最好的感谢方式就是将大学生课外科技活动更好的开展下去。比赛不可能年年获奖, 但通过数学建模教学活动对同学们的能力培养及素质教育所起的作用是有目共睹的: 一周之内, 在教师的指导下, 学习一门计算机语言; 几天的业余时间, 三人合作完成一篇论文, 初步解决一个实际问题, 这在以前同学们是想也不敢想的。“一次参赛, 终身受益”, 已成为所有学生的共识。三人一组所做的建模论文, 使学生们真正懂得了社会需要的不仅仅是他们要有扎实的数学功底, 优秀的学习成绩, 更需要有宽广的知识面, 应用数学知识创造地解决实际问题的能力以及团队精神, 合作意识, 从而激发了进一步学好数学的热情以及解决实际问题的强烈愿望。这又岂是什么奖项能够代表的呢?

大连大学不断融合、改革、发展的风雨历程中, 自强不息的“连大精神”给我们心中注入了坚强的生命底蕴。我们今天的成果绝非仅得力于参赛技巧, 而在于“连大精神”对师生产生强大的吸引力和凝聚力。

回望走过的历程, 我们充满激情, 展望未来的岁月, 育人之路漫漫而修远。为了培养新一代具有创新能力的大学生, 我们将继续执著前行。

### 6.1.2 谈谈我的数模路

我是西北工业大学理学院的数模指导教师肖华勇, 带队参加数学建模比赛已经有 17 个年头了。作为我个人来讲, 我非常喜欢数模。我喜欢数学中的各种模型, 喜欢设计算法和编制程序去求解一个个的问题或建立的模型, 也喜欢对许多实际问题甚至日常的生活问题利用数学建模的方法去思考。就连我特别喜欢的乒乓球我都喜欢从建模的角度去考虑如何提高技术和比赛水平。从某个角度讲, 多年的数模思维, 建模已经深入到我的血液里去了。

当然, 我也特别喜欢带队参加比赛, 也喜欢在课堂上讲解数模。我先后到西安许多高校去讲解过数模, 特别擅长详细讲解数模赛题, 提高了不少学校的参赛水平。在讲解中, 我不但讲解模型的建立过程, 而且编制程序或使用软件详细演示计算求解过程, 使学生清晰的明白每个结果是如何计算出来的。学生都特别喜欢这种十分实在的讲解方法。





在数学建模方面，经过多年努力，我也取得了一些成绩。2008年参与全国数模竞赛的征题成功，有幸成为该年C题命题人。总结多年教学和竞赛的经验，我独立编写了数模方面的教材《基于MatLab与Lingo的数学实验》、《实用数学建模与软件应用》、《统计计算与软件应用》等教材。到2014年为止，我指导的队共获得美国大学生数学建模竞赛特等奖1项，一等奖8项，二等奖8项；获得全国数模竞赛全国一等奖9项，二等奖3项，省一等奖12项。在这些年间，我和队员们遭遇过不少失败，同时也享受过很多成功。下面我结合三个方面谈谈我的数模路。

### 1. 选择数模队伍

作为教练，我谈谈自己选择数模队员的一些经历和感受。记得很多年前，我们整个教练团队是集体培训队员，培训完后按照合适的方式将队员每三人组成一个队。那么哪个教练带哪个队呢？我们采用的是抽签的方式来决定，你抽到哪个队就带哪个队；有时候运气好，抽到一个特别强的队你就走运了，比如2001年我第一次带国际数模竞赛，我就抽到一个能力很强的队。那年我们做的是ICM，当时ICM是两个教练带一个队，我和另一个教练做搭档，我们俩也配合得特别好，刚好我擅长做方向性的决策，搭档擅长细节思考。而队员呢？建模的人能充分懂得教练意图并建好模型，负责计算的队员计算得又快又好，负责写作的是一个女生，特别能写，能很快懂得模型和计算结果，并用英文漂亮地表达出来。结果那年我们队的论文只写了13页，却拿了个国际一等奖。

然而这样的运气只偶尔才有，我从1997年开始带队，到2002年在全国竞赛中一直没有拿过全国奖，总是在省一等奖上徘徊。那时候觉得，拿一个全国奖真的是太难了！另外在带队中，我也发现队员的能力参差不齐，他们也需要特别的培训和参赛经验的积累才能提高参赛水平。怎么才能突破自己，带队拿上全国奖呢？我觉得办法只有一个，一是提高自己的水平，二是提高队员的水平；单靠抽签选到队员后进行训练的方式是远远不够的。于是我给负责数模的老师建议大家可以自由选拔队员，并培训自己的队员，并可以一直带着多次参赛。这个建议后来逐步被负责老师和教练们接受了，之前抽签选队员的方式被废除了；教练可以根据自己的眼光自由选拔合适的队员，并培养出自己的参赛队伍。在这种新的机制和自己的不懈努力下，我所带的参赛队取得了相当好的成绩。

这样教练们有时候在上课过程中发现有优秀的苗子就动员来参加数模。有一次我给数学上概率课，要求学生对一个比较难算的问题做一个计算机模拟，一个同学上台把他的思路清晰地讲了出来，给出了计算机模拟的结果，并且结果也正确。我当时一下觉得这个学生特别适合做数模的编程，他不但有数学思想，而且程序实现能力很强。我就直接动员他来参加数模，结果他跟我做过两次数学建模，竞赛中表现的能力很强，最后拿到了一次全国一等奖和一次国际一等奖。后来毕业后读研，参加研究生数模竞赛，也拿到了全国一等奖。研究生毕业后，这位同学到上海一家软件公司做软件开发了。在上课过程中，不管是给工科学生上公共课，还是给数学系学生上数学课，教练们都喜欢给学生讲讲数模和数模竞赛，这大大激发了学生对数模的兴趣，因此我们学校参加数模竞赛的同学越来越多，





数模竞赛的氛围越来越浓。

后来,我们的教练团队逐渐形成了比较系统的选拔队员的方式。我们学校每年五一都举行校内数模竞赛,参加的人数达到近四千人。说明我们学校学生对数模竞赛的热情是很高的。我们选拔队员首先要求队员参加校内竞赛拿到一等奖,然后有比较高的数学成绩,国际竞赛还要比较好的英语成绩。当然偶尔对成绩不太好,但有特殊才能如编程特别好的队员可以例外。队员报名后,再组织统一的选拔考试,报名做编程的同学就参加编程考试,老师出一道编程题,同学们就现场上机编程,按照做出正确答案的顺序给出排名顺序,供教练挑选编程队员。对参加建模的队员,则给出几个问题供参赛者建模,三小时后交卷,查看选拔者能做出多少东西,供教练选择队员时参考。参加国际竞赛时对写作队员的选拔,是给一两篇科技论文,参加选拔者尽自己能力翻译,规定的时间内完成的翻译量及翻译的水平,作为教练选拔写作队员的参考标准。当然,队员以前是否参加过数模竞赛,如全国数模竞赛,国际数模竞赛,或者电工杯数模竞赛,数模挑战赛等,在教练选拔队员时也显得十分重要。因为数模竞赛不但要求有能力,也希望有参赛经验,拿过奖的更代表水平。好多教练带的队员跟着他们做过多次竞赛。我有一次带一个队就参加过两次全国竞赛和两次国际竞赛,结果拿到一次全国一等奖,一次全国二等奖,两次国际一等奖。队员越做越有经验,教练也省了好多心。这就是队员和教练之间的默契和配合。另外在带队中,我们一个教练通常带两个队,老队员和新队员一块训练、讨论和竞赛,这对新队员也起了一个很好的培养作用。以老带新,也让新队员得到很好的成长。新队员往往第一次参赛拿到省一等奖,第二次参赛就可以拿到全国奖了。

## 2. 赛前指导

选拔好队员后,我们通常先进行基础培训,由各教练分别讲解不同的建模基础知识,如线性规划、回归分析、微分方程、图论与应用、软件使用、数据拟合等,培训时间选择在周末。基础培训完后,就进行赛题练习。这时候将所有的队分成几个大组,每组由四到五位教练组成,每组训练队有10个人左右,由一位教练选择一道合适的赛题,这道赛题可能是以前全国赛题(国际赛就采用以前的国际赛题)或其他学校赛题目或教练自己出的题目,每个队拿到题后,利用一周内的课余时间查找资料,相互讨论,建立模型和编程计算,最后提交论文,并在统一的时间集中在一起,由每个组上讲台报告,坐在台下的教练和同学可以提出质疑,进行进一步的讨论。各教练对自己队伍情况不满意的,可以让他们再进一步完成该论文,以便达到训练效果。

这样在整个培训中,既有基础知识的培训,也有类似真实竞赛的实战训练。特别是赛题培训过程中,各教练对自己队伍论文不满意的,可以提出修改意见,进行反复修改,队员通过这样一种训练,完全知道怎样写好论文,到了真正的竞赛,队员就不怕写作。只要建好模型并编程计算出来,就能把论文写好。

在这种良好地选拔机制和培训机制下,我们的许多教练带的队伍都获得了国际竞赛和全国竞赛的大奖,取得了丰硕的成果。



### 3. 赛后交流

参赛完后，组织队员交流讨论，总结竞赛中的得失。每年竞赛结束后，我们学校所有教练都会聚在一起，开一个小范围的研讨会，总结竞赛中的经验与教训，并讨论下一年数模活动的开展。对全国竞赛，每年国赛后都有一个全国研讨会，教练会去参加研讨，回来后和队员再仔细研究讨论赛题，分析竞赛中哪些方面做得好，哪些方面缺失了，总结经验可以为下次参赛做准备。这样把做过的赛题分析得更清楚，不知不觉就提高了队员的参赛水平，下一次竞赛就可以做得更好。记得2011年我带的队参加国际数模竞赛获得了特等奖（Outstanding Winner），这在西北地区20多年来是第一次，学校很重视，在我们学校自己召开的数模总结会上，特意让我介绍了指导经验，然后大家进行了深入地交流讨论，进一步研究国际竞赛的特点。而我们的队员，也有不少院系请去介绍竞赛经验，让新参加和准备参加竞赛的同学获得许多竞赛经验和知识。

以上是我作为多年数模指导老师的经验和心得，欢迎和各位数模教练以及数学建模爱好者交流。

## 6.2 获奖者的获奖之路

一个成功的数学建模团队，不仅需要优秀的数学建模指导教师，更需要优秀的数模参赛者。作为数学建模竞赛的主人翁，数模前辈的经验显得格外有价值。每一位成功者的背后，都有一段令人心酸的奋斗故事。本节的主人翁为2013年全国大学生数学建模竞赛IBM SPSS创新奖得主周登岳和2014年美国大学生数学建模竞赛O奖以及Frank Giordano Award获得者熊风。

### 6.2.1 成长比成功更重要

各位读者，大家好！我是北京理工大学的周登岳，2013年国赛到现在已经时隔一年了，其实我一直想写这样一篇文章，但却一直拖着没写，一方面是因为繁重的课业，另一方面也是因为自己文笔不好，写东西出来让大家笑话。直到来了德国，几个老师联系我的时候，我才骤然想起那一段无法忘却的青春，所以非常感谢各位老师给了我这次机会，而我也想借这次机会不仅讲讲这一次比赛的经历，更想与大家分享这几年我的成长，我所走过的路。

我刚刚接触数学建模竞赛要追溯到大一下学期，那时学校要组织校内选拔赛，其实我本来也没有打算参加，后来也是因为之前课外德语班上认识的一位女同学（孔垂烨，外号叶子）给我打的电话，她说她们组已经有两个女生了，还差一个人，问我想不想参加，我一听，哎，这个组另外两个都是女生啊，所以我就抱着“可以多认识几个女生”这么一个很简单的想法开始了我两年多的数学建模之旅。

当时我们的队长潘登（外号爬爬）非常要强，对我们要求也特别严格，每周都会给我们开会，规定一部分内容让我们必须学会，然后下次开会的时候直接拿来分析问题，每周





的压力都很大，我经常一个人呆在教学楼里看书到很晚。不过也多亏了爬爬队长，那半年也是我进步最快的半年，数学建模更成为了我生活的主旋律，每天除去上课其余时间基本都在做建模，从排队论到马氏链模型，从蒙特卡罗仿真到神经网络，几乎所有数学建模常用的工具我们都学了一遍。还好我们队参加校内选拔赛的时候成绩确实还不错，当时全校有 200 多个队，我们队 3 次校内赛的总评是全校第 7 名，前 6 名都是大二的，前 10 名只有我们一个队是大一的。

也许正是校内赛成绩的理想让我们看到了希望，于是我们定下了一个目标，全国一等奖，当时在我们的眼中，即使拿全国二等奖也不算是成功。也正是在这个目标的推动下我们三个人几乎拿出了整个暑假来准备比赛，每天都在反复地看各种模型、各种算法和各种优秀论文。2012 年 9 月 10 日，那是我们组第一次参加正式比赛，当时我们队的分工是爬爬主要建模、叶子编程、我是主要负责写作，我们可以说是“势在必得”。当时我们选的题目是葡萄酒的评价一题，简单地说就是根据葡萄酒的一些指标，比如说色调、澄清度、纯正度等建立一套评价体系对葡萄酒进行评价与分级。但一切没有我们想象的那么顺利，第一天我们建立了一个我们自我感觉比较漂亮的模型，但模型特别复杂，非常不好求解，当时我们连续刷了两个夜，72 个小时我们只睡了 8 个小时，一直在用 MatLab、Excel、SPSS 编程、处理数据，光 SPSS 结果文件就生成了一百多个，原本我们幻想这样的付出可以为我们带来回报。可是我们错了，或许也是因为我们水平有限，第一、二问做的过于复杂以至于耗费的时间过多，导致最后一问草草收尾，那次国赛北京理工大学一共有 72 个队伍拿到了市级以上的奖，我们队什么奖也没有拿到，这对于定位于国赛一等奖的我们无疑是一个巨大的打击。

国赛之后，爬爬因为个人原因离队了，于是我就承担起了建模的责任，我们补充了新队员并开始准备 2013 年的美赛。我们选的是一个热学的问题，烤布朗尼蛋糕，尽管我们做好了一切可以做的准备，模型、程序、论文，我们把每一个环节做到了可以做到的最好，其实即使到现在那篇论文也是我感觉我们完成的最好的论文之一；可是最后我们得到的还是参与奖……直到现在我清楚地记得得知结果的那天晚上叶子一边哭一边跟我讲她从来没有学过建模的同学第一次参赛就拿到了美赛 M 奖，而我们为什么都一年了还什么也没有，我的心里其实也特别难受，我一边强忍住眼泪一边安慰她……一整年夜以继日的付出，两次大赛都酱油飘过，而当初和自己一块儿开始搞建模的小伙伴基本都已经有了奖项，我想换谁心里都不会好受。

“还要不要继续参加？”美赛后叶子问我。我说：“要，无论如何也要争一口气，不然对不起过去的一整年。”就这样，我们继续踏上了征程。

我们首先总结了过去一年比赛的经验教训。我们认真分析了优秀获奖论文和我们所写的论文，并反复讨论，发现一到大赛就颗粒无收的尴尬结果并不是因为我们掌握的建模知识不够，也不是因为我们所建立的模型不科学，而是主要在于两方面原因：

### 1. 心态问题

前两次参加大赛的过程中，无论赛前还是赛中，我们都把结果看得太重，甚至认为比





赛就是为了获奖，我们的目标定位也没有放在如何完成一篇优秀的论文上，而是直接以几等奖来量化，我现在还记得2013年美赛比赛期间我们自我感觉过于良好，不止一次地YY我们的论文万一获得了O奖怎么办，现在想想都觉得2333。这种心态直接导致的后果是我们忽略了过程，我们没有拿出所有精力去把论文中的每一个细节做到最好，而是花了大把的时间和精力来模仿特等奖论文的行文、表达方式甚至插图等相对次要的事情上。这就好比一个美丽的泡沫，虽然一刹花火，但却一触就破（额……好像是歌词），看上去很像是特等奖论文但却没有与之相对应的严谨的论述与推导，又怎能得到评委的认同。

### 2. 写最合理的论文，而非最复杂的论文

我之前做建模有一个非常错误的观念就是觉得模型越复杂水平越高，别人越看不懂说明论文写得就越高深，其实并不是那样，尤其是对于我们大学生而言，即使平时拿出大部分课余时间学习相关知识，我们所掌握的也都是有限的，甚至很多时候在赛场上用到的知识也都是比赛的时候现学的，即使能在短时间内搞清楚一个复杂理论的基本原理，要想象在该领域专家一样灵活自如地运用这些新学到的复杂公式也显得非常困难，所以我们很难像各领域专家一样把所有情况都计算在内，更无法消除所有误差，很多时候我们只能照抄所查阅到的文献中的解决套路，然后换几个数，而这种生搬硬套式的解决方案往往又会暴露出一系列问题来，或者应用情况不符，或者缺失先决条件，总之归根结底就是不合理！因此这种生硬的论文自然无法得到评委老师的青睐，在这种情况下，如何对手头问题进行简化就显得尤为重要。我的观点是，合理的简化很重要一点就是要量力而为，量力而为不是妄自菲薄，而是对题目的难度、团队的实力、每个人所擅长的模型和工具进行一个合理的分析与评估，根据评估结果将问题简化到一个经过团队的努力可以在3天内进行完整的建模、求解、写作，并可以将论文的细节做到最完美的程度。这样，就不会出现在比赛初期建立了一个无比复杂的模型，但随着问题的求解却发现困难重重想要简化却发现已经无法回头的尴尬局面，就好比画了一张大饼，最后却发现没法吃。因此，也只有在比赛时间、简化后题目的难度与团队的能力契合的情况下才能写出最合理的论文。

总结会之后，我们开始了新的征程，只是，我们的心态不一样了，从原本势在必得的豪气变得“对结果不再看重”，甚至可以说是无所谓了，我们只是因为喜欢建模在准备比赛，也只是想通过做数学建模为以后在自己的专业领域发表论文打基础。而事实上，因为我的专业原因，大二一年我阅读了大量的汽车动力学、车辆仿真相关的论文，也正是对专业领域论文的研读，我从中发现了数学建模的思想与科研的巨大联系，比如轮胎模型的发展过程就与建模思想有着密切的联系。

于是，2013年国赛的备赛阶段我们并没有像上一次国赛那样拿出大块的时间来准备比赛，也没有拿出大量的时间来学习各种模型、阅读各种论文，而只是拿出平时课余的零碎时间来准备比赛，并将主要时间放在了分析上，我们分析了之前接触过的各种题目、曾经阅读过的论文，对之前犯过的错误进行了反思，并思考如果再在赛场上遇到这类题目该如何处理。值得一提的是，在这段备赛过程中我们还把周吕文老师的网络课程的课件和录音进行了细致的研究，不仅把模型和程序全部吃透，还尝试将题目变形并求解。也正是这次





钻研对我们的国赛起到了巨大的帮助，这是后话。

2013 年 9 月，离我们首次参赛时隔一年，我们再一次参加了国赛，和前两次参加大赛犹如打仗一般紧张的气氛不同的是这次比赛我们都非常放松。

第一天上午，那天我和周晨阳（外号人马，那次比赛我们的队长）上午还有金属工艺实习，当时其他参加国赛的同学都直接调课了，但我们俩觉得不差这几分钟（当时太放松了，其实不应该这样==!），于是上午 8 点赛题发布后我们把两个题目迅速看了一遍就去上课了，直到把锤子加工好到中午 11 点多才回来，中间一直是叶子在下载 A 题视频，据说当时同时下载的人太多导致网路比较卡，直到我们回来的时候才下载好。三个人都到齐后就开始分析选题，A 题题目是《车道被占用对城市道路通行能力的影响》，需要我们根据两个交通事故的视频正确估算车道被占用对城市道路通行能力的影响程度，空间方面各个车辆不连续，时间方面针对事故横断面处车辆通过也不连续；B 题题目是《碎纸片的拼接复原》，则要求我们开发一种碎纸片的自动拼接技术，将碎纸片复原，主要是图像处理算法和编程。我们并没有纠结于选题，经过简短讨论后我们就做出了决定——A 题，很重要的一个原因就是，备赛时我们在研究周老师精讲过的论文时曾经系统地总结过交通类的相关问题，无论是 2005 年美赛 B 题的“收费亭最有数量”问题还是 2009 年美赛 A 题的“交通环岛问题”，都属于这类问题，我们对这类问题可以应用的模型和程序都有比较成熟的想法，无论是排队论还是元胞自动机我们都可以迅速上手编程求解。

得出这一结论来，我们几个精神一下子振奋了起来，跃跃欲试的我们开始对 A 题题目进行细致分析，经过分析我们发现，虽然同属于交通类问题，2013 年国赛题目所涉及的情况还是和那两届的美赛有着比较大的差异，首先各车道的车速、车流量不同，而 2005 年和 2009 年的题目都可以假设所有汽车有着相同的车速，其次车的种类不同，这次的视频录像中有着轿车、公共汽车、客车等多种车型，并且不同车型对道路通行能力的影响是不同的。总之，这次的题目要比之前的交通类题目更加复杂，需要对之前总结的模型和方法进行大幅度修改才能够应用于这次的题目。此外，视频的数据挖掘及错误处理和题干中几个关键概念的定义（如实际通行能力）都成为了 A 题的难点。

明确了需要解决的问题之后，我们分工进行有针对性的文献查阅，也是因为我们之前有经验、手头有资料的原因，往届比赛阅读分析文献的时间至少要半天，而这次我们并没有花费太长的时间在文献阅读上，当天下午三点左右就结束了。

然后我们聚在一起讨论大方案，首先我们发现题干给我们挖了好几个陷阱。

第一个陷阱就是实际通行能力的定义，按照文献中对实际通行能力的定义，它只是根据车型、车道宽度和驾驶员等信息对基本通行能力的修正，而基本通行能力只与交通道路的设计标准有关，也就是说，按照大多数文献中的定义，实际通行能力与视频中的车流数据没关系？！如果没有关系那就没有出题的必要了，所以这里的实际通行能力其实是让我们自己去定义的，于是我们根据所查到的资料并结合题目的信息对实际通行能力进行了重新定义，并对视频进行了处理，从而可以用最大通行能力来替代实际通行能力。

第二个陷阱就是比较坑爹的两段交通事故视频，存在太多的 BUG 了，不仅画面各种缺失跳跃而且非常不清楚，赛后就有很多人吐槽过了（在这里给大家送上一篇写得不错的

吐槽文的链接供各位读者娱乐一下 <http://blog.renren.com/blog/269019659/913819340>), 当时刚比完赛看了后也是笑得不行, 不过玩笑归玩笑, 事实是这两段视频的各种 BUG 其实是出题人故意留给我们, 考验我们的应变能力的。我们当时的处理方法是在将缺失错误的视频部分予以剔除的基础上对视频进行分段, 按照视频中每次显示 120m 的时刻为分界点, 将视频分为了六次排队事件, 从而将视频缺失错误的影响降低到最小。从中我们也可以看出, 国赛已经不仅仅考察我们在完整已知条件的情况下处理问题的能力了, 而是逐渐开始考察我们从不完整条件及包含错误的中提取有用信息的能力。

我们第一天剩余时间的工作重点主要在第一、二问上。一、二问从表面上看只需要依据视频中的车流数据对其加以描述就可以, 但作为全国级别的建模竞赛, 我们认为只用文字进行描述是不够的, 打算对视频数据进行细致处理并建模分析, 更加直观地描述问题。

第一步工作是每个选择 A 题的都比较无力但又无法逃避的——对视频中的车流数据进行挖掘, 所谓通行能力, 字面理解也就是单位时间通过的车的数量, 最开始想到的是像数绵羊一般地去数那些通过的车辆, 第一工作量太大, 第二我们总觉得出题方应该不是为了考察我们的数数能力, 于是我们开始想其他的办法。经过商量, 我们决定采用之前学智能车的时候学到的图像处理方法, 即基于像素灰度的跳变式边缘检测与阈值算法, 通过不断对比每一帧每一个像素点的灰度变化来计算通过某一个截面的车辆, 因为有源程序所以进展比较快。这里需要提到的是, 题目给的几个图片附件其实蕴含了极大的信息量, 这正是体现出题者意图的地方。比如其中提到红绿灯交替时间, 很容易发现, 视频中的车辆是一波一波来的, 每一次绿灯放行的时候便是最容易堵车或者堵车情况加重的时候, 因此红绿灯的影响必须要加到模型中去。此外, 下游路口的车辆在不同车道上的分配也是一个重要原因。如何根据视频实际反映的情况以及题目要求, 去提取合理的建模元素并引入到我们的模型中去便是我们首要的任务。

搞定数据挖掘工作后, 我们就对数据进行了细致的处理和分析, 并开始第一、二问的写作, 与此同时我们还开始了第三、四问的建模工作。第三问需要导出一个关系, 经讨论后, 我们决定从我们比较熟悉的排队论和元胞自动机两套解决方案入手对交通情况进行计算模拟。首先, 元胞自动机模型本身便可以相对全面地动态逼真地模拟出整个交通的变化情况(最开始的预期), 另外, 建立元胞自动机的过程便是解决该问题的过程, 因此这个模型的构建是第三问的灵魂。而第四问则运用第三问的模型进行多次模拟就可以得到计算结果。

其实元胞自动机对于从来没有接触过的同学可能会难度比较大, 建模、编程都需要耗费比较长的时间, 但我们在参加周老师网络课程的时候曾经系统地学习过这个方法, 并在赛前运用元胞自动机进行了实战演练, 无论程序还是论文都有现成的, 对于不同的情况我们直接进行修改即可, 所以对于这个方法我们队伍非常得心应手。

确定了三、四问的大方案之后已经 12 点了, 意味着第一天已经结束了, 我们各回各家休息。第二天一大早我们就开始分头工作, 我开始写第一、二问的论文, 人马和叶子则开始讨论元胞自动机的建模细节, 虽说编程是叶子负责, 但是我们需要把我们建立的模型要求告诉她, 而这个要求的合理性需要我们提前预知, 否则造成编程过于困难以至于难以







求解的话将会极大的浪费时间，因此组内每个人都应该对元胞自动机的机制以及编程实现掌握清楚。到现在这个时间，我们的心态越来越放松，可以说我们已经把建模当着类似日常的任务去做，但是放松归放松，效率不能丢。不时的小组讨论是非常重要的，我们每得到一个结论或者大致结构后都会让每个成员知道，为的是保持成员之间的沟通绝对畅通。而现在回想起来这个是非常必要的。

问题三的基本模型构建结束后，叶子就根据我们大致的要求开始对已有的元胞自动机模型进行修改。我们两个则负责查阅相关文献了解各个术语，偶尔我们也会到网上论坛看看大家的思维，为的是拓展一下思路。对于查阅相关文献的事情，作为数学建模比赛的参赛者来说，最先要明白的一点就是，不可能有任何专业性质的文献能够直接拿来建模。一方面，这个比赛毕竟是全国性质的，而题目肯定是某一个行业的具体问题，如果需要专业知识才能建模的话，那么这种不公平性一定是赛事组织方不愿意见到的。另一方面，建模比赛比的就是建模能力，并不是专业能力。所以，对于查阅到的文献，我们最多的用处便是帮助理解题目中的各个专业名词，另外拓展一下思维，不至于陷入自己的小圈子，而对于那些看起来特别有用的论文，则是完全不可以直接带入的。

在问题三建模编程的同时，问题一、二的论文已经写出来初稿了，遇到了很多困难。第一问中，我们对于实际通行能力的确定，在前文的“陷阱”中已有叙述，我们着实费了一番功夫，文献上的定义可以说对我们产生了极大的误导，所谓在基本通行能力上按照各种条件进行修正得到的交通量，一方面所谓的基本通行能力的定义比较混乱，各个文献之间甚至有冲突，另一方面所谓的修正系数更是需要专业知识才能保证准确，纠结了好久我们才决定放弃各种论文里的计算公式，决定按照本身词语的含义进行计算：所谓实际通行能力嘛，就是单位时间内的最大通行量呗，什么才能叫做最大通行量呢，就是车辆一辆接着一辆的连续不断地通过嘛。针对视频，我们很容易发现有时候可以达到，有时候则是空空荡荡等待车的来临。因此便有了我们第一问对视频的不同时间段的分类。对于第二问，开始走了一些弯路，总感觉要建立一个模型去分析，于是费了好大一番功夫仍没有什么好的结果，最后我们只好妥协，决定只根据数据对其进行文字描述。

第三问中，为了辅助元胞自动机的模型，我们在初步建立起排队论模型对问题进行初步探究，描述了各个因素对排队长度的影响，不过总体上还是以元胞自动机作为工作的重心。建立元胞自动机模型，我们需要提取视频中的各种各样的信息。虽然说这次题目中可以说是没有给任何直接的信息，但是一个视频足以让我们反复观看直到做梦都是你在看监控录像……当然提取出大量的信息后还是需要筛选，比如说事故路段出租车的问题，经常会有出租车开到一半停下来，其实这对车道也算是一种短时间的占用，思索再三，再加上时间原因，我们决定舍弃这个因素，仅仅是提一下，本身出租车也算是小型车辆，而且停靠的话大部分时间和大部分位置都是在非拥挤路段。另一个角度，出租车短时间停车也是交通法规允许的，这也可以证明这种行为不会对交通产生大的影响。经过一番考虑，我们初步决定要将车辆类别、车流量的变化、车流量在不同车道的分配、车速、路段下游方向需求等因素加入模型。其中，车流量在不同车道的分配是一个难点，从视频的统计来看，上游的不同车道的来车和下游车辆的分配有很大的差别，在这一点上其实费了好长时

间，但最终看到官方的解析时也并未能让我们完全信服。遇到这一点时，我们给出的解决方案便是：尽可能的按照题目的意图来，但是不要让这个条件太影响模型。这样一来，一方面可以防止模型中忽略重要条件，另一方面也避免了由于一个条件而导致模型的偏离。

然后我们需要考虑元胞自动机的各个因素，首先，车辆类别，这一点可以从查到的文献中得到，对于不同大小的车辆通过换算系数换算成类似标准当量一样的结果，然后这边是题目中  $\text{pcu/h}$  中  $\text{pcu}$  这个单位的来源。对于这个方面，我们后来在网上看论坛贴吧里讨论的时候，也是很无奈，竟然有这么多种换算方法，最开始我们也是吃了一惊，害怕自己的换算方法有问题，不过后来等到我们也查到好多换算方法时，我们便放松了许多。毕竟这么多计算方法，组委会也不可能给出一个标准去限定什么，而最后的官方解析也可以看到，根本没有这方面的很细致的要求。但是换算只是解决前两问的方法，元胞自动机里可没法换算，如何把原来只有一个单位大小的元胞换成两个单位大小的元胞，这个问题说起来容易，其实这是我们在编程方面遇到的第一个耗时比较多的一个方面。不仅仅是一个变两个，另一方面在运动时需要作为一个整体运动，或者也可能是因为还是没有完全重新熟悉这个模型的原因吧，我们三个人都想了一些方法，不过最终还是非常给力的叶子同学搞定了这个问题（第二天解决的）。

其次，车辆在不同车道的分配，这一点对于整个模型的建立非常重要。更何况题目中明显说明了路段下游车辆的分配情况，显然我们应该将这个作为重点观察对象。在万般无奈的情况下，大家一起抱着屏幕把一个一个车辆数完以后，我们发现上游来的车辆和下游车辆的分配比例有很大的差距。这和我们最初的预期截然相反，后来从司机的角度仔细想想，其实完全没必要在一开始就进入正确的车道，更何况可以看到前方有事故，一定是找最不堵的车道先走，过了事故区后再回到需要的车道上。于是我们决定一方面在最开始加入概率机制，使车辆按照统计出来的规律出现在路段最开始处，在通过事故截面后，将车辆变道概率按照下游车辆分配规律进行修正。

下面一个因素是车速，由于交通事故，司机在看到事故后一定会减速，并以极低的速度通过事故横断面，在通过横断面后，又会提速离开。另外，在不同车道上的车速也是不尽相同，内车道的车速显然快于外车道，这一点也可以通过第二问反映出来。元胞自动机里，由于元胞的运动是以元胞的状态切换实现的，是一种非连续的量，针对这个题目我们可以这样表述：元胞所代表的汽车只能从一个格子跳到另一个格子里去，而这两个之间相隔的数量则是我们自己定义的。这就需要根据视频来进行换算了，从视频中估算出车辆的速度以及大致的速度变化，并换算到元胞自动机模型中去。

把这些因素加进去以后，整个元胞自动机便初具雏形了，我们也是相当兴奋。当然，以上说的这些内容并不是我们在第一天就搞定的，其中有很多曲折、多次修改和否定，我们大部分都是边做边改（也许这是一个不错的方法），不断修正最终才达到一个好的模拟效果。这里提到了一个好的模拟效果，所谓如何验证一个好的模拟效果，当然是要和视频进行比对，而比对的结果如何以一种绝对有说服力的方式展现出来呢，我们想到了显著性检验，利用 SPSS 能够很容易进行这个操作：将建立出来的模型进行模拟仿真，从排队长度、达到特定排队长度所需的时间点两个方面进行检验。当然模型并不是一次性通过验证的，





我们后来又对好多系数进行了匹配修改,才可以达到比较逼真地模拟效果。通过验证以后便直接可以拿来用在第四问了,类似多次试验求取平均值一样的那么轻松加愉快。

上面主要是建模分析过程,可以说整个过程贯穿始终,期间我们也进行了许许多多的其他工作,例如查阅文献、编程处理数据、做统计表格、反反复复一遍又一遍的看视频……怎么说呢,这其实是一个特别痛苦的过程,而且当时我们对自己所做的也没有很大的把握。

到第三天该有的模型、仿真都已经做好,剩下的工作就是论文的写作,总体进行得也比较顺利,最后一天晚上虽然刷夜了,但并不是特别紧张,凌晨4点左右就最终定稿了。

论文打印并提交后,与之前赛后相同的是,仍然感觉五味杂陈,但和前两次大赛不同的是,内心已经没有了获奖的希冀与渴望,更多的是我们团队完成了一次出色的配合,这就够了。后来别人问起来感觉能拿什么奖啊,我都会笑着回答,“争取拿到北京市二等奖吧。”不过我还是猜到了开头却没有猜到结尾。

在那个不眠之夜,我接到了队友的来电,说是不眠之夜其实那天晚上我根本就没法睡觉,因为那几天我正在襄阳参加2013年全国大学生方程式汽车大赛,当时纯电动赛车的电池系统每次充满电都需要十多个小时,白天比赛,只能在晚上充电,所以一到晚上就得待在赛场上。跑题儿了,言归正传,当接到电话得知我们的论文以北京理工大学第一名的成绩报送到全国阅卷的时候,我和小伙伴们都惊呆了,虽然当时比赛时的感觉不错,但之前比赛的经历告诉我们,往往自我感觉越好的时候结果越差,因此对不抱希望的我们来说,这个结果远远超出了预期,激动的我本来就没法睡觉的夜晚更加睡不着了。再后来我们被通知去中科院数学所参加特等奖答辩,当天晚上刷了一夜把答辩PPT做出来,第二天我们三个身着正装给几个老师讲了一个多小时,讲完后一个教授跟我们说,看到这篇论文让他很惊讶,因为他觉得这篇论文不像是三个本科生完成的,就是数学专业的研究生在三天时间内也很难做到,听了教授的话后我特别感动,因为我们的论文得到了老师的肯定。

最后,我们的论文拿到了首届IBM SPSS创新奖及全国一等奖,真的无法用语言描述当时知晓结果时的心情。再后来,我们团队参加了2014年的美赛,没有做任何准备直接参赛就拿到了M奖。比赛往往是这样的,希望越大,失望越大,越放得开,越能得到意想不到的结果。

最后回到这篇文章的标题,“成长比成功更重要”,这句话是我高中的时候最喜欢看的一本书的名字。我之所以以这句话为标题,是因为我觉得我这几年所走过的建模之路正是这句话的真实写照,当我们过分渴望成功的时候,往往不能得偿所愿,而当我们放下对成功的奢望,专心走好脚下的每一步时,成功又会不约而至。

如果说成功是一颗钻石,那么成长就是一条充满荆棘的道路。很多人为了那一颗钻石而患得患失,却不知享受自己不断超越阻碍奔向成功的成长之路,其实在追求成功的过程中,最重要的并不是“打败别人”,而是做一个“最好的自己”。当我们用心品味了我们在成长中的点点滴滴,所有的成功都是在其中,随其后,随之而来的。

我的团队的照片如图6-2所示。





左一：周登岳，左二：孔垂烨，右一：周晨阳，右二：金海（指导教师）

图 6-2

### 6.2.2 如何准备美国大学生数学建模竞赛

我是华中科技大学的熊风，曾经写过一篇关于数学建模的经验贴《两年数模路》，发布在校苑数模论坛上。不过那篇文章主要是讲自己的心路历程，并没太多准备数学建模竞赛的干货。应校苑数模论坛之邀，我结合几次数学建模比赛的经验，来谈谈怎么准备数学建模竞赛，尤其是美国大学生数学建模竞赛。

#### 1. 一些必要的科普

这一部分主要讲数学建模国赛与美赛的区别。虽然对于数模老队员来说，这些应该是常识性的内容；但考虑到不少人对美赛并不是太了解，这里还是科普一下。

我参加过一次国赛，两次美赛。成绩分别是 2012 年国赛国家二等奖，2013 年美赛 ICM 一等奖，2014 年美赛 MCM 特等奖。所以相对国赛来说，我对美赛更为熟悉一些。总结一下美赛的几个特点，一己之见，欢迎拍砖。

##### ◆ 美赛的准备周期比国赛短，参加门槛要低

在我们学校（华中科技大学）要想参加国赛需要通过数模基地的选拔，参加为期一个暑假的培训，时间成本较大；而参加美赛就没这个要求了，原则上只要报名就可以。所以，国赛的准备周期一般是整个暑假，而美赛的准备周期就短得多（如果你在平时上课的时候不准备的话）。美赛的时间一般是春节前后，所以当你结束一个学期的所有考试后，剩下来专心准备的时间一般还不到一个月，在有些年份甚至只有两个星期。比如在 2013 年的时候，当我结束最后一门考试时，离美赛开始只有 10 天。

因此，如果你想充分准备美赛，建议在国赛后就开始准备。因为在临近美赛的时候，





你会把更多时间花到准备大学期末考试上；即使有些年份美赛的开始时间比较晚，你也会受到春节的影响。所以我的建议是尽早开始准备。

而美赛的参赛门槛确实很低，美国数学建模组委会基本没任何限制，只要你们队交 100 美金的参赛费就可以参加。甚至一些高中生就开始组队参加了，曾经也有高中生参赛队实现了斩获 Outstanding 的壮举。

◆ 美赛的获奖比国赛要更容易

总的来说，数学建模竞赛是一个相对容易获奖的比赛。对于不少人来说，一般两三个月时间的认真准备，就能够从新手进阶到国赛一等奖的实力；美赛一等奖相对来说更容易，身边很多没经过数模训练的同学，只是临时组个队去参加也拿到了美赛一等奖。而相比起来，其他的一些比赛（比如 ACM），获奖实在要难太多。

我说这些并非贬低数学建模竞赛获奖的含金量，只是想说数学建模本质上是一个门槛比较低的比赛。几乎所有的理工科学生都可以无障碍地参加，大家不要把这个比赛想得过度困难。

下面仔细分析一下美赛的获奖比例。美赛的结果分为几种：

Outstanding Winners	通常翻译为特等奖
Finalist Winners	通常翻译为特等奖入围奖
Meritorious Winners	通常翻译为一等奖
Honorable Mentions	通常翻译为二等奖
Successful Participants	通常翻译为三等奖

一般来说，只要提交完整的、没跑题的论文，就至少是 Successful Participants。也就是说只要你比赛坚持下来了，就至少有三等奖。每年 Successful Participants 的比例大概有 50%~60%。至于 Unsuccessful Participants 的情况，是非常稀少的，我在身边的人中从未见过。

而二等奖 Honorable Mentions 的比例大概是 30%~35%，一等奖 Meritorious Winners 的比例大概是 10%~15%从这个比例就可以看出，美赛获奖确实比国赛容易一些。在国赛中，国家一等奖和国家二等奖的比例加起来也没超过 8%，而美赛一等奖的比例很多时候就超过 10%了。

所以很多人诟病数学建模美赛获奖的含金量，因为太容易获奖了。从比例来说，这些诟病其实是有道理的。Honorable Mentions 的直译是荣誉提名奖，在很多比赛的奖项设置里，三等奖之后才是 Honorable Mentions，但在中国竟然被翻译成二等奖；Successful Participants 的直译是成功参赛奖，但在中国竟然被翻译成三等奖。所以拿了 Honorable Mentions 就吹嘘自己获得了国际二等奖，这是非常招黑的。

所以我想对每个打算认真参加数学建模美赛的同学说，请至少把目标定为 Meritorious Winners。这也是一个认真准备就可以实现的目标，并不困难。至于更上一层的 Finalist Winners 和 Outstanding Winners，每个题目都只有几个获奖者，难度比 Meritorious Winners 大了很多。要取得这样的成绩还是非常需要运气的，个人觉得不必强求。

### ◆ 美赛比国赛更灵活，更适合自由发挥

一般国赛的题目是这样的：一个长长的 Word 文档，一个或者几个庞大的 Excel 表格。它一般有好几个问题，需要你针对每一个问题严格作答；它给你提供了大量的数据，需要你你对这些数据进行处理。

而美赛的题目风格完全不一样。美赛的题目分为 MCM 和 ICM。MCM 翻译为美国大学生数学建模竞赛，对应着 A 题和 B 题，题目就一两百个单词，不会提供任何数据；ICM 翻译为美国大学生交叉学科竞赛，题目比较长，相当于一篇英文论文，而且往往会提供部分数据和一些参考文献。

这样对比，你会发现国赛的题目比较具体，比较精确；美赛的题目比较抽象，需要自己去拓展，去发挥。

## 2. 赛前准备

### ◆ 组队

一支参赛队一般由三个人组成，比较理想的情况是有人擅长建模，有人擅长编程，有人擅长英文写作。这也是传统的数学建模比赛分工。但是在正式比赛中，分工往往不会这么死板。主要负责建模的同学可能也需要编程来处理一些数据；主要负责编程的同学可能也需要写一下论文来分担队友的工作量。所以，我的建议是，每个人都尽量让自己全能一点，一定不能抱着“不在其位，不谋其职”的态度。不是自己负责的部分就一概不管，这种态度是相当有害的。即使自己能力很弱，即使自己是抱大腿的，在比赛过程中也一定要有担当，不能老想着把麻烦的事推给队友。我们队在比赛的过程中效率高的原因，很大一部分是每个人都很有责任心，都想着尽力帮队友分担任务。这种主人翁意识能让团队的战斗力更强，使比赛过程轻松愉快。

组队是一个很重要的环节，可每个人都受自己环境和人脉的限制，很多时候都无法按照心意组一支自己理想的队伍。所以，我觉得在这里写太多经验也没多大意义。我只强调一点，队友可以没有经验，可以能力弱一些，但一定要有认真的态度。数学建模比赛是一个需要参赛期间全身心投入的比赛，如果队友无法做到全力以赴和全神贯注，绝对是一个大坑。我自己组队的原则就是队友不需要是高手，但一定要靠谱。所以希望大家组队的时候一定要找靠谱的人，在这个基础上再看能不能找到大神。

### ◆ 建模

在这里，我要强调数学建模比赛一个很重要的特点：有广度，无深度。因为没深度，所以门槛低，三个综合素质比较高的人组队，即使没经过任何数学建模训练，也能做出一个不错的结果。他们可以在比赛过程中查找资料，自学需要的模型和算法，写出像样的论文。因为有广度，即使是那些 Outstanding 得主甚至数学建模老师，也无法通晓比赛中可能涉及的模型和算法。

所以，即使你是从未参加过比赛的新手，面对比赛也不用心虚，“兵来将挡，水来土掩”碰到没学过的东西不用怕，你可以现学嘛；即使你是拿过无数大奖的老手，也不要自满，因为你掌握的东西仅仅是数学建模中的沧海一粟。





既然数模的范围这么广，那我们还需要准备吗？答案是当然需要。

如果你要系统地准备数学建模，我最推荐的资料是网上一套叫《数学建模算法大全》的教程，随便搜索一下就可以找到。这套资料我也不知道出自何处，但确实写的非常好，深度对于准备比赛来说比较合适。每一章都讲得比较浅，不难看懂，适合入门，同时也有 MatLab 和 lingo 代码，适合学习。

如果你的准备时间很短，没有时间系统地学习那个教程里面的各种算法。那我给个建议，数学建模中的问题通常分为这么几个大类：优化、预测、分类、评价。最好保证能大概清楚每一个大类里有哪些常用的模型和算法，并且至少熟悉其中的一种。

在建模方面，我推荐 DC 学长的一篇博文《数模国赛前一天总结》。DC 学长获得过国赛的国家一等奖，这篇博文是他赛前准备工作的清单。如果你们队准备到这个程度，那应该也可以对比赛胸有成竹。

#### ◆ 编程

编程是各路大神各显神通的环节了，比赛没有任何限制，你可以使用你熟悉的任何编程语言。但是依照我的经验，在绝大多数情况下，MatLab 足以解决问题。所以数模中编程语言首选 MatLab，世界公认加默认。分析一下 MatLab 的优缺点以及它为什么适合数学建模比赛。

优点：

(1) 各种函数和工具箱。这点简直不能更赞，基本上数模中所需要用到的成熟算法都能在 MatLab 中找到函数和工具箱。比如拟合之类的问题，使用 MatLab 工具箱简直轻松愉快。所以，熟练使用 MatLab 会大大提高数学建模中的编程效率。

(2) 画图功能非常强大。在数学建模比赛中，需要画大量的图形。用 MatLab 再合适不过了，很容易用 MatLab 画出各种高大上的图形。

缺点：

效率低，计算慢。我觉得这点在数学建模比赛中甚至算不上缺点。因为比赛对程序时限要求不高，只要能出结果就行了。即使 MatLab 花一分钟才能算出 C 语言几秒就算出的东西，也没多大关系。

但还是有极少数的一些情况，可能用别的语言或者工具比较合适。比如对于一些其他格式的数据（带有文字）进行读入或者转换的时候，我更习惯用 C++ 或者 python。比如规划问题，虽然 MatLab 的工具箱已经足够强大；但 lingo 这样的工具更为简便。学会使用 lingo 大概只需要几个小时，所以还是很建议学一下。比如一些统计方面的问题，用 SPSS 和 Eviews 更加方便。这些也建议一个队伍中的三个人至少有一个人会，毕竟这些都比较容易学，并且技多不压身嘛。

另外在比赛的过程中，可能一些题目需要用到比较小众的软件。遇到这种情况不要慌，要相信自己的自学能力。比如图 6-3 所示为犯罪关系网络示意图，是 2012 年 ICM 华科 Outstanding 论文中的图形，如果你比赛中需要画这样的图，你打算怎么画呢？这显然不是用 MatLab 和 office 系列工具可以有效解决的，需要自己学习一些比较小众的软件。

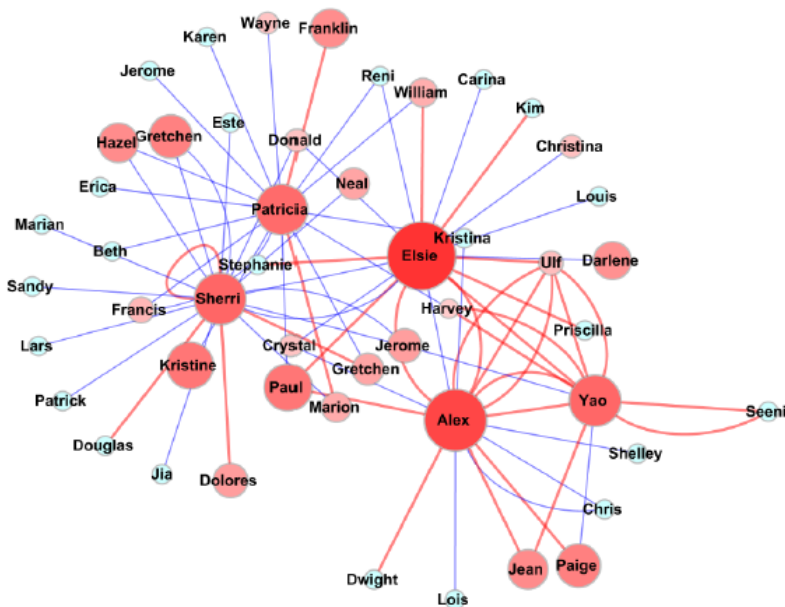


图 6-3

总结一下，我觉得数学建模中负责编程的同学需要会的技能。

基本：能熟练使用 Word、Excel、Visio 这种基本软件进行数据处理和画图。熟悉 MatLab 的基本编程，可以写程序处理数据和画各种图形。

进阶：熟悉 MatLab 的一些常用函数和工具箱，熟悉 lingo。在数学建模中遇到拟合、回归、规划以及其他调用函数就能解决的问题，能够迅速用 MatLab 这种工具有效地求解。

高级：遇到求解方面的问题，能通过各种手段（不管是自己写代码还是用 MatLab 工具箱或是用小众软件）迅速解决。

另外，对于常见的模型和算法，建议在平时准备好代码，等到比赛时再到处找代码、写代码，那就比较费时间了。好的习惯就是平时将各种代码搜集整理好，并且保证自己调试过。这里说一个血的教训，我们队在国赛的时候需要用到主成分分析这种方法。但我平时准备的主成分分析代码是从网上下载的，并没有用过；在比赛中使用的时候发现代码出现了问题，在这个本应该几分钟解决的问题上耗费了几个小时。

总的来说，数学建模中的编程并不需要非常强的编程能力，很多时候甚至不需要自己写多少代码，更重要的是能熟练使用各种工具迅速解决问题。我个人认为有过 ACM 经验（即使还没到铜牌水平）的同学承担数学建模中的编程任务已经绰绰有余了。

最后还是推荐一下 DC 学长一篇讲数学建模编程的博文，文章里还列了代码清单，很值得参考。

#### ◆ 写作

因为评委们看不到你们辛辛苦苦推导公式的过程，也看不到你们精妙的代码，他们能看到的只有呈现在眼前的一篇文章，所以论文的写作非常重要。

我见过很多实力很强的队伍，模型和求解都非常漂亮，但是论文没写好，最终结果不



如意。这是相当遗憾的一件事。所以一定不要忽视论文的写作，尤其是摘要的写作。

第一个问题，美赛中用 LaTeX 还是用 Word？我觉得都可以。虽然熟练 LaTeX 之后效率会很高，但我觉得如果没时间学 LaTeX 或者不想学 LaTeX，用 Word 也没问题。我们队就是老老实实用 Word。

第二个问题，关于论文的格式和排版。这些东西一定要提前准备好模板，不要拖到比赛的时候去做，太浪费时间了。队伍里那个负责写作的同学，一定要非常熟悉 LaTeX 或者 Word 的操作，尽量不要在比赛的过程中遇到不会解决的排版问题。负责排版的同学一定要细心，最好有一些强迫症，这样才能保证精益求精。我们比赛拿到 Outstanding 很大程度是因为有一个强迫症队友。他对细节的追求可谓苛刻，他无法容忍任何一个地方排版不好看，哪里多了一个空格，哪里行距看着不舒服，哪里的公式格式不太对，他都会精心改正。在数学建模比赛中，这是非常繁琐的工作，但是确实需要人去做。

第三个问题，关于英语论文的写作和表达。大家不要过于心虚，我觉得写好一篇数模论文并不需要太高超的英语能力。说实话，我们队的英语能力就不是太好，写英文论文的过程那是相当煎熬，对着有道翻译，写得非常慢。一些数模里常用的专业术语，建议平时要积累。如果一个人的英语水平有限，最好每个队友都检查一下，确认没有语病，语句是通顺的。

如果有时间，最好在赛前每天练习英语阅读和写作。阅读的材料可以是以前的优秀论文，收集一些比较好的表达句式和方法。总之论文的写作非常非常重要，有时候条理清晰、语言流畅就足以让你的论文获奖上升一个档次。

#### ◆ 一些建议

赛前需要准备好的一些东西：

- 比赛报名：请一定确认报名成功；
- 比赛场地：三个人在一个房间，便于交流；网络流畅，室内温暖，可以通宵使用（一般最后一个晚上都要通宵的）；
- 自己的电脑：准备好各种工具软件；可以翻墙；有登陆数据库查论文的途径；
- 论文写作的模板：Word 或者 LaTeX；
- 常用算法的代码（见前文）。

### 3. 比赛过程

比赛的时间长达四天四夜，所以比赛的节奏是很重要的。这也是队长应该重点控制的。

首先，我不建议拼得太凶，比如有些队四天只睡 10 个小时之类的，我觉得太凶残了，也没太大必要。美赛还是更看重方法和效率的，很多问题不是靠打疲劳战就可以解决的。我建议的作息方式还是正常作息，只不过最后一天要通宵。我们队在 2014 年美赛的过程中，每天大概都是 8 点左右起床，上午 9 点开始干活，晚上 9 点半收工回寝室，回寝室后再忙下自己这部分的工作，12 点前睡觉。

#### ◆ 比赛第一天

上午最好完成选题的工作，最迟也要在晚饭前确定。很多人选题太过谨慎，患得患失，





耗费了太多时间。个人觉得不是太有必要，我们队选题完全凭感觉，看哪题顺眼选哪个。但是有一个原则：一旦选中，就再也不悔改了。不然会浪费大量时间，影响士气。

下午这段时间属于头脑风暴，每个人各自根据自己的理解去尝试建一下模型，查一下资料。然后约定个时间进行讨论，经过三个人的确认，最终得到一个初步的方向。

晚上就是具体的建模了。比较理想的进度是能在第一天晚上就完成大部分建模的工作，对之后的工作有方向上的把握。

### ◆ 比赛第二天

继续建模，最好能在中午确定大概的模型。接着就需要分工了，如果题目需要查数据查资料，让一个人查数据；同时写论文的同学开始写论文，写整体的构架；编程的同学也可以初步进行编程求解了。

晚上最好能出一个初步的结果，得到部分数据，对自己的模型有一个大概的评估。如果结果实在不好的话，可以考虑调整自己的模型。这一天我建议一定要把建模部分的工作全部完成，如果建模没搞定，要适当熬一下夜。

### ◆ 比赛第三天

编程求解的工作基本应该完成，应该能得到所有结果的初步数据。而论文的写作也可以同步进行。这一天很关键，如果求解结果很 nice 的话，说明进度控制得不错。

这一天最好不要熬夜，因为最后一天是肯定要通宵的。建议这天好好睡一觉来迎接最后的战斗。

### ◆ 比赛第四天

可以说是最关键的一天。比较理想的进度是吃午饭之前完成所有的编程求解部分的工作，然后下午开始做模型部分的分析。我觉得分析是一个很重要的部分，可以让自己的论文提升一个档次。如果没思路，至少可以做一些敏感性分析，深入地分析自己的模型，并且画一些图；如果有灵感，想想如何增添亮点。

比较理想的进度是在晚上 12 点前完成论文的初稿。

接下来，一个队伍要花大概两三个小时的时间写好摘要。摘要这一关实在是太重要了。要知道，那些拿到 **Successful Participants** 的论文，全部是在摘要这一关被刷下去的。可以这样说，摘要写得不好，肯定只有 **SP**；摘要写好了，至少也是 **H** 奖。所以写摘要实在是论文写作中最需要投入的事了。由于我们队的英语都不是太好，所以摘要是一起写的，每一句都经过了反复修改。最后得出完整稿的时候，我们还重读了很多遍，做出了很多修改。

把摘要写完，基本就可以得到最终稿了。剩下的就是一遍遍读论文，检查论文的错误了。最终确定无误后就可以提交了。恭喜你们，完成了整个美赛之旅。

当然，以上只是我们的进度参考。在比赛的过程中，什么事都有可能发生；灵活制订自己队伍的策略，才是王道。

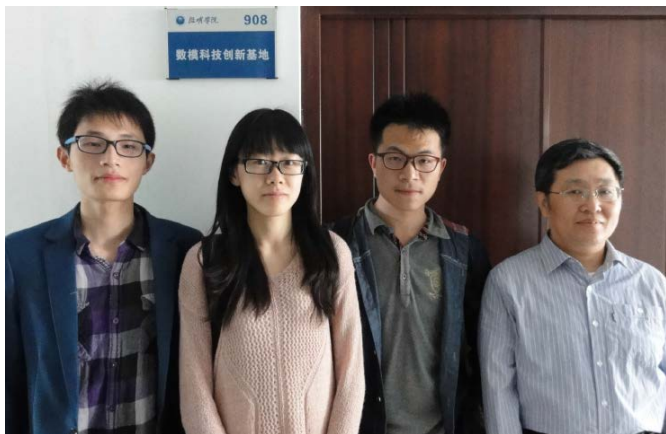
## 4. 总结与感想

我对美赛的情感比较复杂，它带给了我很多，但同时也让我看到了数学建模竞赛的局限性，让我更清楚自己以后要走的路。Anyway，我自己的数学建模竞赛之路走到了终点，



虽然有遗憾，但总体上还是满意的。希望后继的人们能享受数学建模的旅程，取得自己满意的成绩。


我的团队照片如图 6-4 所示。



左起依次为：丁文超、李晶玲、熊风、韩志斌（指导教师）

图 6-4

## 6.3 本章小结

无论是指导教师的多年经历，还是优秀获奖者的奋斗历程，都验证了一句话“没有经历过失败，哪有资格接受成功！”。我们的数模之路终将结束，但是数学建模带给我们的难忘经历以及宝贵的能力将永远伴随着我们，在今后的学习和生活中，请大家始终记住下面这段话“有些事不是看到了希望才去坚持，而是坚持了才会看到希望！”。 



## 第7章 互联网资源

在信息高速传递的时代，数学建模参赛者要具备从网络获取准确信息的能力。随着数学建模在全国的持续普及，各类数学建模网站和论坛如雨后春笋般呈现在大家面前。高校也已经建立或者打算建立校内的数模网站，便于校内学生使用，同时宣传学校的数学建模活动。

那么当前国内知名的数学建模网站都有哪些呢，哪些网站更值得同学们经常逛逛呢？相信很多数模爱好者都曾有过类似的烦恼，这里梳理了一下当前知名的数学建模网站，并着重介绍校苑数模网（<http://www.mathor.com>）和交叉学科数学建模学生论坛。

### 7.1 校苑数模论坛

数学建模参赛队应该在赛前掌握一些常用的建模网站，比如全国大学生数学建模的官方网站 <http://www.mcm.edu.cn/>，美国大学生数学建模的官方网站 <http://www.comap.com/>以及各个赛区的官方网站（目前全国各赛区的官方网站都建立在赛氪网 <http://www.saikr.com>）。比赛的官方网站往往只具备发布赛题以及获奖名单等功能，这对于参赛者平时的交流而言是不方便的。知名的数学建模信息和资源的网站有以下四个：校苑数模网 <http://www.mathor.com>，中国数学建模网 <http://www.shumo.com/>，数学中国 <http://www.madio.net> 以及赛氪网 <http://www.saikr.com>。这些网站中有大量数学建模的知识、往届获奖者的经验分享以及优秀论文、数学建模常用软件以及数学建模活动。接下来，本节着重介绍校苑数模网。

#### 7.1.1 历史介绍

要说校苑数模网，还不得不提到它的前身——校苑资源网，校苑资源网已经死掉了，如果去搜索引擎搜索还是可以看到它的影子的。2008年，大二上半年的我，接触了数学建模，发现数学建模是非常有意思的。因为当你真正用心去做这件事的时候，你发现很难，但是一旦你入门了，你会发现，不知不觉中获得了很多知识和见识。最主要的是，数学建模极大提高了参与者的学习能力和意志力。当时发现，参加数学建模过程搜集资料非常麻烦，要与更多的人交流非常局限。进而构想，不仅仅是数学建模，大学生获取其他学习资源的渠道也是捉襟见肘的。于是，我产生了做个基于大学生获取资源的网站的想法，通过网络联系到了大连市内一个做网站的工作室，以1500元的价格请他们给做了个网站，另外，利用开源的论坛程序给我们搭建了个BBS，当时不知道开源的程序都有什么，所以还额外





支付了 500 元作为开源社区的搭建。刚开始只是基于兴趣，参与者除了我之外还有我的同学任立峰、李蔓蔓和已经毕业的学姐范倩倩，记不清当时是怎么“忽悠”他们的了，总之是忽悠进来跟我做这个大学生资源网。网站运营了 2 年，考研用户占据了大多数，当时注册用户达到 50 多万，对于在校的我们来说是非常兴奋的。兴奋的同时灾难也随之而来，不是特别懂服务器技术的我们面临着服务器总是宕机的困扰。服务器不够用，要升级，升级就需要钱。我四处奔走求援，在奔走过程中，不负责任的 IDC 将我们的服务器转租了，数据都丢失了，真可谓晴天霹雳。就这样团队散伙了 2 个月之后，我重新聚集大家做起了现在的校苑数模网。希望能在数学建模方面帮到更多的大学生。

校苑数模网成立后，很多校苑资源网的热心网友了解到了校苑数模转过来一批铁粉。从无到有，我们做了大概半年校苑数模，网站就具有了非常大的影响力。今天，校苑数模已经发展成为全国知名的数学建模社区，这也证明了我们当初的决定是正确的。

### 7.1.2 校苑简介

校苑数模是一个专注于数学建模“学习”、“交流”的平台，致力于数学建模知识分享、经验总结以及竞赛交流。网站始终坚持以“服务”为宗旨，努力满足广大数模爱好者的需求，受到了广大高校师生的认可与支持。

校苑数模网是由清华大学数模爱好者、中国科学院博士以及互联网企业中的数模爱好者共同管理，全国各地高校师生共同维护的一个大家庭。网站在坚持“公益数模”的前提下，举办了大量特色活动。

#### ◆ 校苑数模高校巡回讲座

2014 年校苑数模正式举办“校苑数模高校巡回讲座”活动，该活动的目的是促进高校间数模爱好者的交流，平衡高校间的优质数学建模资源。从 8 月底至 12 月中旬，校苑数模先后在清华大学、北京航空航天大学、西南财经大学、成都信息工程学院、武汉理工大学、西北大学以及西南民族大学 7 所高校举办此活动，先后邀请了中国科学院力学所博士周老师、2014 年美赛特等奖以及 Frank Giordano Award 华中科技大学熊风、西北工业大学交叉学科数学建模学生论坛嘉宾姜杰出、2014 年美赛特等奖获得者以及 SIAM 奖得主西南民族大学谢永意等。这一活动在全国范围内引起强烈反响，各大高校的老师联系论坛，希望论坛能够在他们学校举办一次类似活动。

#### ◆ 校苑数模高校交流会

2015 年校苑数模论坛受华北电力大学（保定校区）的邀请，与华北电力大学师生做了一次交流。此次交流会由三位同学担任主讲嘉宾，分别在美赛参赛经验、数模与科研的关系两个方面做了交流。此次交流会通知发布后，广大高校的数学建模负责老师通过 QQ、微信以及邮箱等与校苑数模取得联系，咨询交流会情况。为此，校苑数模论坛在 2015 年正式开启“校苑数模高校交流会”活动。该交流会旨在分享不同高校数学建模的培养方式，以及交流数学建模的发展和数模活动的举办。



- ◆ 国赛和美赛赛前公益培训

高教社杯获得者以及美赛特等奖获得者在国赛或者美赛前，无疑会受到广泛关注，为此，校苑数模在国赛和美赛前邀请了往届高教社杯获得者以及美赛特等奖在线分享自己的数模经历，传授参赛技巧。

### 7.1.3 展望

校苑数模是一个年轻的数模网站，更是一个充满朝气与活力的网站，校苑数模团队成员都曾感受过刚刚接触数模时的无助与迷茫，经历过参赛失败。这份感受和经历正是指引校苑数模发展的风向标。目前越来越多的网站开始关注校苑数模，越来越多的师生都在期待着校苑数模更上一层楼。

校苑数模郑重承诺：以人为本是网站的核心价值观，服务大家是我们的座右铭，你们的需求永远都是我们努力的方向！最后以校苑寄语结束校苑数模的介绍，“有些事不是看到了希望才坚持，而是坚持了才会看到希望”！

## 7.2 交叉学科数学建模学生论坛

为了推动数学建模发展，探讨交叉学科数学建模方法，提升学生数学应用能力，由西北工业大学主办的第一届“交叉学科数学建模学生论坛”（Student Forum on Interdisciplinary & Mathematical Modeling, SFIMM）于2014年5月31日—6月1日在古都西安成功举行。

“2014 交叉学科数学建模学生论坛”中，国内数学建模专家和国内20余所高校的大学生共200多名代表参加这次为期两天的学术论坛。北京理工大学叶其孝教授，东南大学朱道元教授，复旦大学谭永基教授、蔡志杰教授，西安交通大学周义仓教授，北京邮电大学贺祖国教授，西北工业大学聂玉峰教授、彭国华教授、李玉龙教授、刘佩进教授出席并做报告。MCM/ICM主办方COMAP执行主席Solomon Garfunkel博士致信论坛组委会祝贺此次学生论坛举办。

第一届交叉学科数学建模学生论坛有效地实现了以学生为主体的数学建模学术交流，推动了交叉学科数学建模研究的深入，为广大学生提供了参与交流研讨的平台，有助于树立崇尚学术的风气、营造学术交流的氛围，突破竞赛思想屏障，是由数学建模竞赛向数学建模研究转变的一次有效尝试。

### 7.2.1 创建论坛的初衷

以数学建模竞赛为核心而蔚然形成的数模文化，早已形成了西北工业大学的一道奇观，并因这道奇观的盛大壮观、与众不同而享誉国内、蜚声国际。因此，全球最具影响力的大学生学科竞赛——数模竞赛，成为西北工业大学规模最大、参与学生最多、受益面最广的大学生课外科技活动。在西北工业大学，每年有近5000名学生参加校内数模竞赛，本科生参赛率达80%以上，每年有数百名学生参加全国和国际大学生数模竞赛。或许不了解数模





的读者会追问：为什么要开展大学生数学建模竞赛？大学生数学建模竞赛魅力何在？将数学建模引入教育过程的终极意义何在？这些问题，在 2013 年西工大国际数模竞赛最高奖“三连冠”之际，西工大的数模教练和数模学生就已经悉数解读。

兴趣驱动创新。西北工业大学无论是本科生还是研究生在数学建模方面浓厚的兴趣推动下，经常开展有关数学建模的经验交流、圆桌讨论、主题演讲等学术交流活动，极大地推动了数学建模活动的发展，同时也推动了数学建模方法在研究生科研中，特别是交叉学科中的应用。甚至有不少校内竞赛赛题背景都是数模学生在交流之中获得灵感，进而深入研究，通过与数学建模指导教师探讨、试算，最终加工形成具有一定研究背景且难度适中的赛题。这也是“问题驱动的数学建模”的一个体现。

发起和推动本次学生论坛的五位核心成员，来自西北工业大学材料学院、航天学院、机电学院的朱斌（大四）、孙林（研二）、李仲谋（研一）、虞天晔（大三）、舒毅潇（研一），皆有多年的数学建模竞赛经历，并屡次获得从全国数模竞赛一等奖到国际数模竞赛特等奖的优异成绩。五位同学因为数学建模而认识，因为对学术研究的兴趣成为朋友。在交流中发现，本科生在基础教育中遇到了这样一些问题：

（1）学生受到的数学教育脱离应用实际。数学应该是工科大学生的最重要的一门基础学科，然而我们发现大部分同学“不知道学了数学有什么用，学了数学能做什么”的困惑，也即“学而无趣”、“学而无用”的现象。我们相信这样的现象不单单仅存在于数学学习中，其他的专业课程同样面临这样的挑战。

（2）大学生没有太多接触到科学研究的机会。很多高校都以建设研究型大学为目标，可是大多数本科生却不知道研究是什么，做研究的过程是什么，而最关键的是做研究的乐趣是什么。这使得学生把科学研究看得太过神圣，高不可攀。因此到了毕业面临选择时，大部分的同学自然地避开了研究的这条路，而即使选择了做研究继续深造，其中很多人也是浑浑噩噩不知其所以然。

（3）学生之间学术交流氛围并不是很浓厚。大部分本科生专注于学习和考试，而很少和其他人进行经验交流、观点讨论，更谈不上学术交流；在研究生中，各院系之间的专业研究领域存在相似或相同的学术问题，但是硕博研究生仍然是闭门造车为主，缺乏有效途径进行更广泛的专业领域内外的科研动态、研究方法等的有效共享和交流。

（4）数学建模团队缺乏继承性。作为一个理论应用、创新实践的学术交流平台，我们看到数学建模团队每年培养了一大批优秀的数模人才，取得了优异的成绩。然而，每一届的新老队员之间经验方法交流不足，传承性、连续性太差，导致学校每年在新队员的培养中存在大量低效的重复工作，而无法通过学生一届届地传承，形成内部经验方法交流的有效循环。

作为“兴趣带动数学学习，问题驱动数学研究”的最直接方法，数学建模在一定程度上解决了“学习与应用脱节”的问题。但是，一直以来，更多的学生是在参加竞赛，却没有了解到什么是真正意义上的数学建模研究。学生通过选拔参赛、竞赛结束直至获知竞赛结果，均缺少有效的竞赛赛题讨论，而对赛题的后续探究也是甚少。而目前存在的交流会也均以教师为主。如此现状，使得竞赛后老师、学生之间的互动需要进一步加



强,也亟待一个交流的平台,使得作为数学学习与应用的数学建模能够得到有效的交流和探讨。

同时,数学建模是用数学方法来解决实际问题的一项实践活动。在这过程中所遇到的问题极少是纯粹数学方面的问题,基本上都涉及到交叉学科的知识。在以往的数学建模竞赛赛题中,也涉及到诸如环境科学、电信工程、机械制造、生物医学、材料科学、金融管理、信息科学等领域,因此有必要结合跨学科这一主题,开展交叉学科数学建模学生论坛。

现今的数学建模方向学术会议中少有以学生为主的学生会议,因此,开展这样一个学生论坛,促进了学生主体在交叉学科中的探索和数学建模领域的研究。以研讨会的形式交流数学建模竞赛的方法,避免了竞赛的时间局限性,使得学生进一步深入研究竞赛中提出的问题,是探索大学生创新能力培养的新举措,与全国组委会组织的数学建模赛题后续研究工作的目的是一致的。同时,论坛邀请的机械、电子、计算机等交叉学科的优秀研究生和本科生做交叉学科中的数学建模专题报告,也是对跨学科、跨领域创新过程中遇到的问题的一种新的解决方法;同时能够达到开阔知识视野,发掘学术兴趣和增强学术功底的目的;能广泛涉猎各个学科领域,优化知识结构,提升综合科研素质。

于是,“2014 交叉学科数学建模学生论坛”的举办提上日程。

为什么选择交叉学科数学建模为主题?

(1) 数学建模为学生创造了一个学以致用用的平台。作为“兴趣带动数学学习,问题驱动数学研究”的最直接方法,数学建模一定程度上解决了“学习与应用脱节”的问题。通过数学建模,学生利用平常所学习到的数学思维、数学知识、数学工具,针对现实中的问题进行:分析问题,交流讨论,求解模型,分析结果,归纳结论。让学生能够意识到所学习到的知识不是抽象的,而是可以具体地应用于现实中的。

(2) 数学建模为学生提供了一个力所能及,通过实践来增强科研认知的机会。数学建模毫无疑问是研究的一种雏形,而这种雏形恰恰是学生力所能及的。通过参与数学建模,学生有机会了解到科研的基础和方式:如何查阅阅读文献,如何分析求解问题,如何撰写科技论文,从而增强学生的科研能力。同时,通过完成数学建模,学生能够从自己的实际动手过程中发现自己的乐趣,从而对真正的研究产生兴趣。

(3) 数学建模为交叉学科学生奠定了一个可以交流的基础。在西北工业大学浓厚的数学建模氛围中,每年校内竞赛就有超过 5000 人参加,在三天的数学建模中,参与的学生因为对数模题目倾注了心血,所以一定有可以交流的想法和交流的欲望。数学建模作为交叉学科的凝结点,在三航、机械、材料、电子等工业工程领域都有应用,同时在工程实际问题研究中也必然会被涉及到。因此以数学建模为桥梁,可以为学生开辟出一条不同专业领域学术交流的途径。

(4) 学生论坛为数模团队创造了一个实现传承的契机。一年一次的学生论坛以及平时的学术沙龙,为数学建模团队新老队员之间提供了日常的经验总结、方法交流的机会,以此实现数模团队的传承。





## 7.2.2 交叉学科数学建模学生论坛五大亮点

开展学生论坛，希望达到两个目的。其一，为广大本科生、研究生了解科学研究、提供学术交流的平台。对于本科生，主要是通过数学建模接触科研、认识科研；对于研究生，主要以学术科研中的问题进行相互交流。其二，促进西北工业大学学子保守的、闭门造车式的、“小农”学术思想的革新。而本次论坛形式多样、交流充分，呈现出以下五大亮点：

(1) 开创性地实现了学生自发组织、自主参与的学术研讨。论坛全程以学生为主体、由学生唱主角，让学生真切感受和认识到数学建模的魅力、数学建模的思维及数学建模在工程领域的重要价值。

(2) 突破竞赛思想屏障，实现从竞赛到真正的研究。经历本次论坛的探讨与交流，让与会学生进一步明白，“当我们能用数学语言来描述某个问题的时候，其实就是一个数学建模的过程”，“数学建模是科研的雏形，当人们运用数学模型来解决工程问题时，就进入科学研究”。

(3) 学生参与命题的创新做法。西北工业大学每年向全校师生和社会公开征题，有很多大学生和研究生积极为校数模竞赛提供赛题。近三年，校内数模竞赛的6道题目中就有5道是西工大的本科生和研究生命题的。

(4) 问题征解的创新形式。论坛专家报告中，长江学者、飞行器动力学专家李玉龙教授和发动机研究专家刘佩进教授与学生们进行了交叉学科建模方法的探讨，并提到工程上的一些数学问题，希望通过数学建模发布出去。两位专家教授以问题征解的形式提出亟待解决的工程问题，给与会学生以很大的触动。

(5) 促进大学数学教育改革，传承数模团队文化。数学建模竞赛是现代数学教育教学改革的重要突破口，数学建模是拓展提升现代数学创新教育的有效路径，而此次会议以学生为主体，探讨交叉学科的数学建模方法，正是现代数学创新教育的重要尝试。同时，不同年级不同专业、甚至不同学校学生交流经验，碰撞思想，也使得数模团队文化得以传承。

## 7.2.3 总结与展望

回望论坛从设想到具体实施，不就是数学建模中对问题的分析、建模、求解的过程吗？当论坛结束后，撰写总结报告，不正是后续模型评价吗？当然，初次举办论坛，主办方经验不够丰富，模型不够成熟，模型的细节也不够完善，还有很多纰漏亟待总结、反思。

本次论坛仅仅只是一个良好的开端，专家们对论坛提出的意见和建议很多，后期还有一系列的工作需要谋划。与会专家这样评价本次论坛：由学生自发组织举办“交叉学科数学建模学生论坛”，是数学建模创新教育的有效实践，在一定程度上突破竞赛思想屏障，是由数学建模竞赛向数学建模研究转变的一次有效尝试。

论坛专家组叶其孝教授评价：“我觉得2014交叉学科数学建模学生论坛非常的好，我想我们国家将来都是这些年轻同志的天下。年轻同学能够独立地开展一些学术活动，而且




组织起来、互相影响，这件事情在我们国内是很了不起的。我觉得西北工业大学的学生，特别是（交叉学科数学建模学生论坛）组委会的这些同学做得非常好，我觉得这件事情很值得在全国推广。”

专家组蔡志杰教授说道：“2014 交叉学科数学建模学生论坛据我所知是国内第一次由学生主办的跟建模有关的学术活动，这是一个非常好的开始。数学建模对学生能力的培养是有目共睹的，但是，很多学生在做了数学建模竞赛以后，对数学建模竞赛题目的后续研究方面的活动比较少，交叉学科数学建模学生论坛在这一方面起到了一个很好的作用，搭了一个平台，使得学生和学生之间、学生和老师之间有了一个互相交流的机会，并且能够从老师和其他学生这里得到对问题更深层次的理解，对问题的后续研究会起到一个比较大的作用。”

参会嘉宾说：“我全程参加了论坛活动，让我认识了多位国内知名的数学专家、评卷老师、数模带队教练、各级数学建模竞赛最高奖获得者，他们对问题求解的缜密思维、追求知识和真理的精神，深深触动了我”，“参会打开了视野，开拓了思维，让我从竞赛中跳出来，使我的认识由先前单纯的数模竞赛上升到科学研究”。

此次论坛的成功开展，定会激励各位热爱数学建模的同仁再接再厉。希望各位数模爱好者继续关注“交叉学科数学建模学生论坛”，我们一道借助数学建模思想，用数学建模精神，开展更深入的交流讨论。

以上只是筹办交叉学科数学建模论坛中的一些想法和实践，还有许多方面考虑的不成熟，在此也热忱地欢迎各位专家老师、各位数学建模爱好者不吝赐教。 





# 参考文献

## 第 1 章参考文献

- [1] 杨光惠, 刘合财. 浅谈数学建模竞赛论文写作[J]. 黔南民族师范学院学报, 2009, (3)73~94

## 第 2 章参考文献

- [1] K.S.Cline.SecretsoftheMathematicalContestinModeling  
<https://www.carroll.edu/kcline/mcm.pdf>
- [2] BrianCamley,PascalGetreuer,BradleyKlingenberg.TheQuestoftheMCM  
[http://www.math.pitt.edu/~rubin/mcm\\_guide\\_colorado.pdf](http://www.math.pitt.edu/~rubin/mcm_guide_colorado.pdf)
- [3] 2015 美国数学建模 LaTeX 模版更新|LaTeX 工作室.<http://www.LaTeXstudio.net/2015-mcm-LaTeX-template-update/>
- [4] HMCmath: HMCICM/MCMinformation.<https://www.math.hmc.edu/mcm/>
- [5] TobiasOetiker.一份不太简短的 LaTeX2e 介绍.<http://www.ctan.org/tex-archive/info/lshort/chinese>.

## 第 4 章参考文献

### 参考文献 (1)

- [1] 杨洪. 图论常用算法选编[M], 中国铁道出版社, 1988
- [2] 卢开澄等. 图论及其应用[M], 清华大学出版社, 1995
- [3] 袁震东, 洪渊等. 数学建模[M], 上海: 华东师范大学, 1995
- [4] 谢金星, 薛毅. 优化建模与 LINDO/LIBGO 软件[M], 北京: 清华大学出版社, 2005
- [5] 薛定宇, 陈阳泉. 高等应用数学问题的 MatLab 求解答[M], 清华大学出版社, 2004

### 参考文献 (2)

- [1] 陆凤山. 排队论及其应用[M]. 长沙: 湖南科学技术出版社, 1984
- [2] 钱颂迪等. 运筹学[M]. 北京: 清华大学出版社, 1991
- [3] 姜启源等. 数学模型[M], 第 4 版. 北京: 高等教育出版社, 2011
- [4] 谢金星, 薛毅, 优化建模与 LINDO/LIBGO 软件[M], 北京: 清华大学出版社, 2005

### 参考文献 (3)

- [1] Cellular automaton-wikipedia, the free encyclopedia. Available at <http://en.wikipedia.org/>





wiki/Cellular\_automaton [Accessed March 1, 2014]

[2] Conway's game of life, November 2014. Available at [http://en.wikipedia.org/wiki/Conway's\\_Game\\_of\\_Life](http://en.wikipedia.org/wiki/Conway's_Game_of_Life) [Accessed October 10, 2014]

[3] Wolfram atlas: Elementary cellular automata: index of rules. Available at <http://atlas.wolfram.com/01/01/rulelist.html> [Accessed October 10, 2014]

[4] Elementary cellular automaton, October 2014. Available at [http://en.wikipedia.org/wiki/Elementary\\_cellular\\_automaton](http://en.wikipedia.org/wiki/Elementary_cellular_automaton) [Accessed October 10, 2014]

[5] 1999 mcm problems. Available at <http://www.comap.com/undergraduate/contests/matrix/PDF/1999mcmProblems.htm> [Accessed October 10, 2014]

[6] 2013 年高教社杯全国大学生数学建模竞赛赛题. Available at <http://www.mcm.edu.cn/problem/2013/2013.html> [Accessed August 10, 2014]

[7] 2013 mcm problems. Available at <http://www.comap.com/undergraduate/contests/mcm/contests/2013/problems/> [Accessed March 1, 2014]

[8] 2013 优秀论文选登. Available at <http://special.univs.cn/service/jianmo/2013lw> [Accessed August 10, 2014]

[9] 孔垂烨, 周晨阳, 周登岳. 车道被占用对城市道路通行能力的影响. Available at <http://special.univs.cn/service/jianmo/2013lw/2013/1112/1000401.shtml> [Accessed August 10, 2014]

[10] Winstrom Luke, Coskey Sam, and Mark Blunk. Shelling tumors with caution and wiggles. The UMAP Journal, 24(3):365–378, 2005. Available at [www.math.washington.edu/~morrow/mcm/uw25.pdf](http://www.math.washington.edu/~morrow/mcm/uw25.pdf) [Accessed March 1, 2014]

[11] Forest-fire model, May 2014. Available at [http://en.wikipedia.org/wiki/Forest-fire\\_model](http://en.wikipedia.org/wiki/Forest-fire_model) [Accessed August 10, 2014]

[12] S.Clar, K. Schenk, and F.Schwabl. Phase transitions in a forest-fire model. Physical Review E, 55(3):2174–2183, March 1997

[13] Peter Grassberger. On a self-organized critical forest-fire model. Journal of Physics A: Mathematical and General, 26(9):2081, 1993

[14] Kai Nagel and Michael Schreckenberg. A cellular automaton model for freeway traffic 2(12):2221–2229. Available at <http://fix.bf.jcu.cz/~bereg/NagelSchreckenberg1992.pdf> [Accessed March 1, 2014]

[15] Yuki Sugiyama, Minoru Fukui, Macoto Kikuchi, Katsuya Hasebe, Akihiro Nakayama, Katsuhiro Nishinari, Shin-ichi Tadaki, and Satoshi Yukawa. Traffic jams without bottlenecks-experimental evidence for the physical mechanism of the formation of a jam 10(3):033001. Available at [http://iopscience.iop.org/1367-2630/10/3/033001/pdf/1367-2630\\_10\\_3\\_033001.pdf](http://iopscience.iop.org/1367-2630/10/3/033001/pdf/1367-2630_10_3_033001.pdf) [Accessed March 1, 2014]

[16] Experiment creates test track jam. Available at <http://www.traffictoday.com/>

news.php? NewsID=3962 [Accessed March 1, 2014]

[17] 堵车有哪些原因? Available at <http://www.zhihu.com/question/20781120> [Accessed March 1, 2014]

[18] Shockwave traffic jam test - video. Available at <http://www.maniacworld.com/shockwave-traffic-jam-test.html> [Accessed March 1, 2014]

[19] Formation of a 'phantom traffic jam' | MIT video. Available at <http://video.mit.edu/watch/formation-of-a-phantom-traffic-jam-8152/> [Accessed March 1, 2014]

[20] Kai Nagel and Hans J. Herrmann. Deterministic models for traffic jams. Physica A: Statistical Mechanics and its Applications, 199(2):254–269, October 1993. Available at <http://www.comphys.ethz.ch/hans/p/156.pdf> [Accessed March 1, 2014]

[21] Benjamin Seibold, Morris R. Flynn, Aslan R. Kasimov, and Rodolfo Ruben Rosales. Constructing set-valued fundamental diagrams from jamiton solutions in second order traffic models. Available at <http://arxiv.org/pdf/1204.5510.pdf> [Accessed March 1, 2014]

#### 参考文献 (4)

[1] Tabu search, December 2014. Available at [http://en.wikipedia.org/wiki/Tabu\\_search](http://en.wikipedia.org/wiki/Tabu_search) [Accessed December 20, 2014]

[2] Simulated annealing, December 2014. Available at [http://en.wikipedia.org/wiki/Simulated\\_annealing](http://en.wikipedia.org/wiki/Simulated_annealing) [Accessed December 20, 2014]

[3] Genetic algorithm, December 2014. Available at [http://en.wikipedia.org/wiki/Genetic\\_algorithm](http://en.wikipedia.org/wiki/Genetic_algorithm) [Accessed December 20, 2014]

[4] Artificial neural network, December 2014. Available at [http://en.wikipedia.org/wiki/Artificial\\_neural\\_network](http://en.wikipedia.org/wiki/Artificial_neural_network) [Accessed December 20, 2014]

[5] Ant colony optimization algorithms, December 2014. Available at [http://en.wikipedia.org/wiki/Ant\\_colony\\_optimization\\_algorithms](http://en.wikipedia.org/wiki/Ant_colony_optimization_algorithms) [Accessed December 20, 2014]

[6] 司守奎, 孙玺菁. 数学建模算法与应用第1版. 北京: 国防工业出版社, edition, August 2011

#### 第5章参考文献

##### 参考文献 (1)

[1] 薛定宇, 陈阳泉. 高等应用数学问题的 MatLab 求解答[M]. 北京: 清华大学出版社, 2004

[2] 谢金星, 薛毅. 优化建模与 LINDO/LIBGO 软件[M]. 北京: 清华大学出版社, 2005

[3] 肖华勇. 实用数学建模与软件应用[M]. 西安: 西北工业大学出版社, 2008

[4] 袁震东, 洪渊等. 数学建模[M]. 上海: 华东师范大学, 1995

[5] 姜启源等. 数学模型[M]. 第4版. 北京: 高等教育出版社, 2011





参考文献 (2)

- [1] Traffic flow - wikipedia, the free encyclopedia. [http://en.wikipedia.org/wiki/Traffic\\_flow](http://en.wikipedia.org/wiki/Traffic_flow)
- [2] 2005 MCM problems. <http://www.comap.com/undergraduate/contests/mcm/contests/2005/problems>
- [3] 2009 MCM problems. <http://www.comap.com/undergraduate/contests/mcm/contests/2009/problems>
- [4] 2009 CUMCM problem. [http://en.mcm.edu.cn/problem/2013/2013\\_en.html](http://en.mcm.edu.cn/problem/2013/2013_en.html)
- [5] 2014 MCM problems. <http://www.comap.com/undergraduate/contests/mcm/contests/2014/problems>
- [6] Traffic simulation - wikipedia, the free encyclopedia. [http://en.wikipedia.org/wiki/Traffic\\_simulation](http://en.wikipedia.org/wiki/Traffic_simulation)
- [7] Microsimulation - wikipedia, the free encyclopedia. Available at <http://en.wikipedia.org/wiki/Microsimulation>
- [8] Kai Nagel, Dietrich E. Wolf, Peter Wagner, and Patrice Simon. Two-lane traffic rules for cellular automata: A systematic approach. 58(2):1425–1437. <http://arxiv.org/pdf/cond-mat/9712196.pdf>
- [9] Ning WU and Werner Brilon. Cellular automata for highway traffic flow simulation. II. [http://homepage.rub.de/ning.wu/pdf/ca\\_14isttt.pdf](http://homepage.rub.de/ning.wu/pdf/ca_14isttt.pdf)
- [10] Peter Wagner, Kai Nagel, and Dietrich E. Wolf. Realistic multi-lane traffic rules for cellular automata. 234(3):687–698. <http://e-archive.informatik.uni-koeln.de/238/1/zpr96-238.ps>
- [11] Modeling traffic flow for two and three lanes through cellular automata. <http://www.m-hikari.com/imf/imf-2013/21-24-2013/martinezIMF21-24-2013.pdf>
- [12] Intelligent driver model - wikipedia, the free encyclopedia. [http://en.wikipedia.org/wiki/Intelligent\\_driver\\_model](http://en.wikipedia.org/wiki/Intelligent_driver_model)
- [13] Longitudinal traffic model: The IDM. Available at <http://www.vwi.tu-dresden.de/~treiber/MicroApplet/IDM.html>
- [14] Arne Kesting, Martin Treiber, and Dirk Helbing. General lane-changing model MOBIL for car-following models. Transportation Research Record, 1999(1):86–94, January 2007. [http://www.akesting.de/download/MOBIL\\_TRR\\_2007.pdf](http://www.akesting.de/download/MOBIL_TRR_2007.pdf)
- [15] Martin Treiber, Ansgar Hennecke, and Dirk Helbing. Congested traffic states in empirical observations and microscopic simulations. Physical Review E, 62(2):1805, 2000. <http://arxiv.org/pdf/cond-mat/0002177.pdf>
- [16] Aaron Abromowitz, Andrea Levy, and Russell Melick. One ring to rule them all: The optimization of traffic circles. The UMAP Journal, 30(3):247–260, 2009. Available at <http://www.cs.hmc.edu/~rmelick/docs/MCMPaper.pdf>

[17] The lane-change model MOBIL. <http://www.vwi.tu-dresden.de/~treiber/MicroApplet/MOBIL.html>

[18] Martin Treiber and Arne Kesting. Modeling lane-changing decisions with MOBIL. In Cécile Appert-Rolland, François Chevoir, Philippe Gondret, Sylvain Lassarre, Jean-Patrick Lebacque, and Michael Schreckenberg, editors, *Traffic and Granular Flow '07*, pages 211–221. Springer Berlin Heidelberg, January 2009. [http://link.springer.com/chapter/10.1007%2F978-3-540-77074-9\\_19#page-1](http://link.springer.com/chapter/10.1007%2F978-3-540-77074-9_19#page-1)

[19] Microsimulation of road traffic flow. <http://www.traffic-simulation.de/>

[20] M. Schreckenberg, A. Schadschneider, K. Nagel, and N. Ito. Discrete stochastic models for traffic flow. *Physical Review E*, 51(4):2939–2949, April 1995. arXiv: cond-mat/9412045

[21] Traffic flow, June 2014. [https://en.wikipedia.org/wiki/Traffic\\_flow](https://en.wikipedia.org/wiki/Traffic_flow)

[22] Doboszczak, Stefan and Virginia Forstall. Mathematical modeling by differential Equations. <http://www.norbertwiener.umd.edu/Education/m3cdocs/Presentation2.pdf>

[23] Mit mathematics | traffic modeling. <http://math.mit.edu/projects/traffic/>

[24] Kenneth G. Courage, Charles E. Wallace, and Rafiq Alqasem. modeling the effect of traffic signal progression on delay. *Transportation Research Record*, (1194), 1988. <http://trid.trb.org/view.aspx?id=302135>

[25] R. F. Benekohal and Joseph Treiterer. Carsim: car- following model for simulation of traffic in normal and stop-and-go conditions. *Transportation Research Record*, (1194), 1988

[26] Should you switch lanes in traffic? | what can scientific models tell us about the world? <http://playingwithmodels.wordpress.com/2010/06/24/should-you-switch-lanes-in-traffic>

参考文献 (3)

[1] McMichael. A.J. Planetary overload: global environmental change and the health of the human species: Cambridge University Press, 1993

[2] Pereira, H.M. et al. Scenarios for global biodiversity in the 21st century. *Science* 330, 1496–1501, 2010

[3] Lavergne, S., Mouquet, N., Thuiller, W., & Ronce, O. Biodiversity and climate change: integrating evolutionary and ecological responses of species and communities. *Annual Review of Ecology, Evolution, and Systematics*, 41, 321–350, 2010

[4] Jackson, S.T., Betancourt, J.L., Booth, R.K., & Gray, S.T. Ecology and the ratchet of events: climate variability, niche dimensions, and species distributions. *Proceedings of the National Academy of Sciences*, 106(Supplement 2), 19685–19692, 2009

[5] Barnosky, A. D., Hadly, E. A., Bascompte, J., et al. Approaching a state shift in Earth's biosphere. *Nature*, 486(7401), 52–58. doi: 10.1038/nature11018, 2012

[6] Kurant, M., Thiran, P., & Hagmann, P. Error and attack tolerance of layered complex





networks. Physical Review E, 76(2), 026103, 2010

[7] Kurant, M., & Thiran, P. Layered complex networks. Physical review letters, 96(13), 138701, 2006

[8] Environmental Performance Index and Pilot Trend Environmental Performance Index, 2012, Release (2000–2010) <http://sedac.ciesin.columbia.edu/data/set/epi-environmental-performance-index-pilot-trend-2012>

[9] Lotka–Volterra equations, [http://en.wikipedia.org/wiki/Lotka%E2%80%93Volterra\\_equation](http://en.wikipedia.org/wiki/Lotka%E2%80%93Volterra_equation)

[10] Britton, N.F. Reaction-Diffusion Equations and Their Applications to Biology. Academic Press, 1986

[11] Anderson, J.D. Fundamentals of aerodynamics (Vol. 2): McGraw-Hill New York, 2001

[12] University of California - Berkeley. "Evidence of impending tipping point for Earth." ScienceDaily, 6 Jun. 2012

[13] 赵慧霞, 吴绍洪, 姜鲁光, 生态阈值研究进展, 生态学报, 27 (1): 338-345, 2007

[14] Tipping Points, <http://www.savetheplanet.org.cn/gb/science/tippingpoints.html>

[15] Scott, J. Social network analysis. Sociology, 22(1), 109-127, 1988

[16] Lifshits, MA. Gaussian random functions (Vol. 322): Springer, 1995 